



## 저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

Ph.D. DISSERTATION

# Exploitation of Visual Relationships for Semantic Image Understanding

이미지의 의미적 이해를 위한 시각적 관계의 이용

BY

DAESIK KIM

AUGUST 2019

Digital Contents and Information Studies  
Department of Transdisciplinary Studies  
Graduate School of Convergence Science and Technology  
SEOUL NATIONAL UNIVERSITY

Ph.D. DISSERTATION

# Exploitation of Visual Relationships for Semantic Image Understanding

이미지의 의미적 이해를 위한 시각적 관계의 이용

BY

DAESIK KIM

AUGUST 2019

Digital Contents and Information Studies  
Department of Transdisciplinary Studies  
Graduate School of Convergence Science and Technology  
SEOUL NATIONAL UNIVERSITY

# Exploitation of Visual Relationships for Semantic Image Understanding

이미지의 의미적 이해를 위한 시각적 관계의 이용

지도교수 곽 노 준

이 논문을 공학박사 학위논문으로 제출함

2019년 8월

서울대학교 대학원

융합과학부 디지털정보융합전공

김 대 식

김대식의 공학박사 학위 논문을 인준함

2019년 8월

위 원 장:	이 교 구	(인)
부위원장:	곽 노 준	(인)
위 원:	서 봉 원	(인)
위 원:	이 원 종	(인)
위 원:	이 민 식	(인)



# Abstract

Understanding an image is one of the fundamental goals of computer vision and can provide important breakthroughs for various industries. In particular, the ability to recognize objective instances such as objects and poses has been developed due to recent deep learning approaches. However, deeply comprehending a visual scene requires higher understanding, such as is found in human beings. Humans usually exploit contextual information from visual inputs to detect meaningful features. In this dissertation, visual relation in various contexts, from the construction phase to the application phase, is studied with three tasks.

We first propose a new algorithm for constructing relation graphs that contains relational knowledge in diagrams. Although diagrams contain richer information compared to individual image-based or language-based data, proper solutions for automatically understanding diagrams have not been proposed due to their innate multimodality and the arbitrariness of their layouts. To address this problem, we propose a unified diagram-parsing network for generating knowledge from diagrams based on an object detector and a recurrent neural network designed for a graphical structure. Specifically, we propose a dynamic graph-generation network that is based on dynamic memory and graph theory. We explore the dynamics of information in a diagram with the activation of gates in gated recurrent unit (GRU) cells. Using publicly available diagram datasets, our model demonstrates a state-of-the-art result that outperforms other baselines. Moreover, further experiments on question answering demonstrate the potential of the proposed method for use in various applications.

Next, we introduce a novel algorithm to solve the Textbook Question An-

swering (TQA) task; this task describes more realistic QA (Question Answering) problems compared to other recent tasks. We mainly focus on two issues related to the analysis of the TQA dataset. First, solving the TQA problems requires an understanding of multimodal contexts in complicated input data. To overcome this issue of extracting knowledge features from long text lessons and merging them with visual features, we establish a context graph from texts and images and propose a new module f-GCN based on graph convolutional networks (GCN). Second, in the TQA dataset, scientific terms are not spread over the chapters and subjects are split. To overcome this so-called “out-of-domain” issue, before learning QA problems we introduce a novel, self-supervised, open-set learning process without any annotations. The experimental results indicate that our model significantly outperforms prior state-of-the-art methods. Moreover, ablation studies confirm that both methods (incorporating f-GCN to extract knowledge from multimodal contexts and our newly proposed, self-supervised learning process) are effective for TQA problems.

Third, we introduce a novel, weakly supervised object detection (WSOD) paradigm to detect objects belonging to rare classes that do not have many examples. We use transferable knowledge from human-object interactions (HOI). While WSOD has lower performance than full supervision, we mainly focus on HOI that can strongly supervise complex semantics in images. Therefore, we propose a novel module called the “relational region proposal network” (RRPN) that outputs an object-localizing attention map with only human poses and action verbs. In the source domain, we fully train an object detector and the RRPN with full supervision of HOI. With transferred knowledge about the localization map from the trained RRPN, a new object detector can learn unseen objects

with weak verbal supervisions of HOI without bounding box annotations in the target domain. Because the RRPN is designed as an add-on type, we can apply it not only to object detection but also to other domains such as semantic segmentation. The experimental results using a HICO-DET dataset suggest the possibility that the proposed method can be a cheap alternative for the current supervised object detection paradigm. Moreover, qualitative results demonstrate that our model can properly localize unseen objects in HICO-DET and V-COCO datasets.

**keywords:** visual relation, graph structure, recurrent network, question answering, graph convolution network, object detection, weakly-supervised learning

**student number:** 2014-24888

# Contents

<b>Abstract</b>	<b>i</b>
<b>Contents</b>	<b>iv</b>
<b>List of Tables</b>	<b>viii</b>
<b>List of Figures</b>	<b>ix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Problem Definition . . . . .	4
1.2 Motivation . . . . .	6
1.3 Challenges . . . . .	7
1.4 Contributions . . . . .	9
1.4.1 Generating Visual Relation Graphs from Diagrams . . .	9
1.4.2 Application of the Relation Graph in Textbook Question Answering . . . . .	10
1.4.3 Weakly Supervised Object Detection with Human-object Interaction . . . . .	11
1.5 Outline . . . . .	11

<b>2</b>	<b>Background</b>	<b>13</b>
2.1	Visual relationships . . . . .	13
2.2	Neural networks on a graph . . . . .	16
2.3	Human-object interaction . . . . .	17
<b>3</b>	<b>Generating Visual Relation Graphs from Diagrams</b>	<b>18</b>
3.1	Related Work . . . . .	20
3.2	Proposed Method . . . . .	21
3.2.1	Detecting Constituents in a Diagram . . . . .	21
3.2.2	Generating a Graph of relationships . . . . .	22
3.2.3	Multi-task Training and Cascaded Inference . . . . .	27
3.2.4	Details of Post-processing . . . . .	29
3.3	Experiment . . . . .	29
3.3.1	Datasets. . . . .	29
3.3.2	Baseline. . . . .	32
3.3.3	Metrics. . . . .	32
3.3.4	Implementation Details. . . . .	33
3.3.5	Quantitative Results . . . . .	35
3.3.6	Qualitative Results . . . . .	37
3.4	Discussion . . . . .	38
3.5	Conclusion . . . . .	41
<b>4</b>	<b>Application of the Relation Graph in Textbook Question Answering</b>	<b>46</b>
4.1	Related Work . . . . .	48
4.2	Problem . . . . .	50
4.3	Proposed Method . . . . .	53
4.3.1	Multi-modal Context Graph Understanding . . . . .	53

4.3.2	Multi-modal Problem Solving . . . . .	55
4.3.3	Self-supervised open-set comprehension . . . . .	57
4.3.4	Process of Building Textual Context Graph . . . . .	61
4.4	Experiment . . . . .	62
4.4.1	Implementation Details . . . . .	62
4.4.2	Dataset . . . . .	62
4.4.3	Baselines . . . . .	63
4.4.4	Quantitative Results . . . . .	64
4.4.5	Qualitative Results . . . . .	67
4.5	Conclusion . . . . .	70

## 5 Weakly Supervised Object Detection with Human-object Interaction 77

5.1	Related Work . . . . .	80
5.2	Algorithm Overview . . . . .	81
5.3	Proposed Method . . . . .	84
5.3.1	Training on the Source classes $D_S$ . . . . .	86
5.3.2	Training On the Target classes $D_T$ . . . . .	89
5.4	Experiment . . . . .	90
5.4.1	Implementation details . . . . .	90
5.4.2	Dataset and Pre-processing . . . . .	91
5.4.3	Metrics . . . . .	91
5.4.4	Comparison with different feature combination . . . . .	92
5.4.5	Comparison with different attention loss balance and box threshold . . . . .	95
5.4.6	Comparison with prior works . . . . .	96

5.4.7 Qualitative results . . . . .	96
5.5 Conclusion . . . . .	100
<b>6 Concluding Remarks</b>	<b>105</b>
6.1 Summary . . . . .	105
6.2 Limitations and Future Directions . . . . .	106
<b>Abstract (In Korean)</b>	<b>121</b>
<b>Acknowledgement</b>	<b>124</b>

# List of Tables

3.1	Comparison results of AP on the AI2D test set. . . . .	33
3.2	Comparison results of IoU on the AI2D test set. . . . .	34
3.3	Comparison results of Recall@K on the AI2D test set. . . . .	34
3.4	Accuracy of Question Answering on AI2D and FOODWEBS . .	37
4.1	Comparison of data types in context and question parts for con- text QA, VQA and TQA. . . . .	48
4.2	Comparison of performance with previous methods and results of ablation studies. . . . .	60
4.3	Results of ablation study about the occurrence flags. . . . .	66
5.1	Comparison of quantitative result of different feature combination.	92
5.2	Comparison of quantitative result of different attention loss bal- ance and box threshold. . . . .	93
5.3	Comparison of quantitative results with prior works . . . . .	94



# List of Figures

1.1	Illustration of visual relationship. . . . .	2
1.2	Illustration of the tasks addressed in this dissertation. . . . .	5
3.1	Illustration of generating relational knowledge from diagram . .	19
3.2	Overview of the unified diagram parsing network (UDPnet) . . .	20
3.3	Comparison of the vanilla GRU and the proposed DGGN . . . .	24
3.4	Specific explanations of update and retrieve steps in DGGN . . .	25
3.5	Qualitative results of diagram graph generation . . . . .	31
3.6	Statistics of activation value of update gate on AI2D test sets . .	39
3.7	Mean of activation values of update gate . . . . .	40
3.8	Additional qualitative results on diagram graph generation . . .	42
3.9	Additional qualitative results on diagram graph generation . . .	43
3.10	Additional qualitative results on diagram graph generation . . .	44
3.11	Additional qualitative results on diagram graph generation . . .	45
4.1	Examples of the textbook question answering task and a brief concept of our work . . . . .	47
4.2	Analysis of contexts in TQA and SQuAD datasets. . . . .	50
4.3	Overall framework of our model in TQA . . . . .	51

4.4	Illustration of f-GCN. . . . .	54
4.5	Self-supervised open-set comprehension step in our model. . . .	58
4.6	Additional examples of SSOC steps. . . . .	59
4.7	Qualitative results of text-type questions without visual context.	67
4.8	Qualitative results of diagram-type questions. . . . .	68
4.9	Additional qualitative results on text-type question with visual context. . . . .	71
4.10	Additional qualitative results on text-type question with visual context. . . . .	72
4.11	Additional qualitative results on diagram-type question without visual context. . . . .	73
4.12	Additional qualitative results on diagram-type question without visual context. . . . .	74
4.13	Additional qualitative results on diagram-type question with vi- sual context. . . . .	75
4.14	Additional qualitative results on diagram-type question with vi- sual context. . . . .	76
5.1	Illustrations of description of an object and Manually annotating time for three tasks. . . . .	78
5.2	Overview of our algorithm for WSOD. . . . .	82
5.3	Distribution plot of categories of objects in the training set of HICO-DET. . . . .	83
5.4	Overall network architecture of the proposed algorithm. . . . .	85
5.5	Comparison of predicted attention maps trained by different inputs	97
5.6	Unsuccessful results. . . . .	98

5.7	Illustrations of attention maps for verb “Hold” . . . . .	101
5.8	Illustrations of attention maps for verb “Ride” . . . . .	102
5.9	Illustrations of attention maps for verb “Carry” . . . . .	103
5.10	Illustrations of attention maps for verb “Sit on” . . . . .	104

# **Chapter 1**

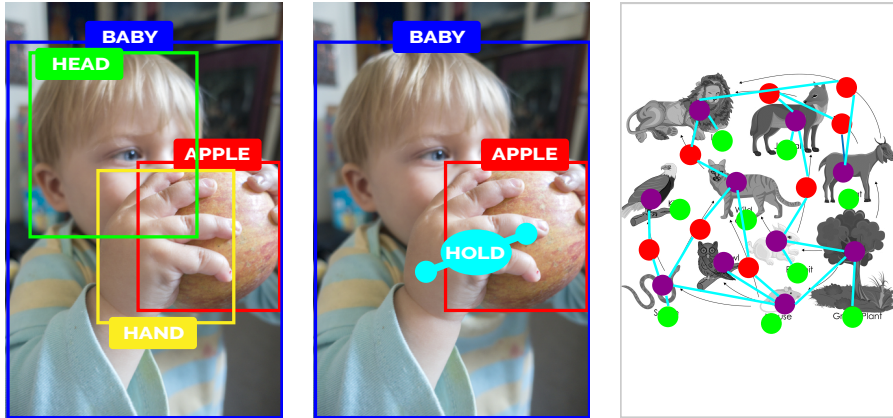
## **Introduction**

Understanding an image has been one of the most fundamental goals in the field of computer vision. Since what humans see provides them with information that is fundamental to survival, understanding human vision may be one of the most important tasks of artificial intelligence. Moreover, visual information can be used for applications in various industries to solve practical issues.

Recognizing constituents in images has been actively researched since the publication of early computer vision literature. Algorithms have been challenged to extract meaningful features from images of pixel structure for various applications. For instance, several researches have attempted to extract low-level features.

Today, the deep learning approach has enhanced optimization of parameters to capture sufficient features of various input types. Therefore, recent methods have made significant improvements for extracting features to recognize objective instances, especially objects and poses, in images.

However, in order to deeply understand a visual scene, a system requires semantic image knowledge rather than simple recognition. Humans mainly con-



(a) Object detection

(b) visual relationship in a natural image

(c) visual relationships in a diagram

Figure 1.1: Illustration of visual relationship. In contrast to a leftmost figure which shows detected objects, rest figures present visual relationships among objects in a natural image and a diagram. In this dissertation, we are interested in generating visual relationships and applying practical problems.

vert visual inputs into abstract semantic information through visual contextual information.

This visual contextual information can imply various concepts, such as the geometrical layout, the taxonomy of objects and relationships between objects. In addition to objective facts about images, information inferred and constructed from prior knowledge can provide sophisticated semantics that can contribute to solving widespread applications, such as chatbots and robotics. In this dissertation, we focus on visual relationships between instances such as the object and the human on images.

Visual relationships are usually represented between objects in natural images and more intentionally expressed in figures such as diagrams. Since the

diagram contains rich illustrations including text, visual information, and their relationships, it can be worth to infer semantic information. Therefore, this research studies visual relations in diagrams as well as in natural images.

The intention of this dissertation is to propose novel methods that can be used to construct a graphical structure of relations in diagrams and exploit semantic contexts with natural language to solve a multimodal problem. Moreover, we propose a new object recognition scheme to efficiently train a system to identify unseen objects using visual relation in natural images.

First, we propose a novel end-to-end model to construct relation graphs of diagrams. A new module is also proposed to efficiently propagate information using graph structure. We investigate data flow inside the module to analyze the contributions of our methods.

Second, we suggest a new method to solve the most complex QA problem using relational knowledge acquired from the previous approach. For solving the multimodal issue efficiently, we fuse visual context and textual context into an integrated graph structure. Moreover, we propose a novel self-supervised scheme to pre-train parameters with out-of-domain data splits.

Third, a novel, weakly supervised scheme using transferred relationship knowledge is proposed to optimize an object detector using natural images. We also propose a module to localize bounding boxes to transfer contextual information. Our method can successfully train existing object detectors such as Faster-RCNN on unseen object classes with weak supervision with low annotation cost.

The remainder of this chapter is organized as follows. In Section 1.1, we define the problems that are addressed throughout this dissertation. Each of the three tasks in this dissertation is separately explained. Then, the motivation of

three tasks is discussed in Section 1.2. Challenges of the three tasks that must be overcome are enumerated in Section 1.3. Contributions of the methods proposed in this dissertation are discussed in Section 1.4. Finally, an outline of the dissertation is given in Section 1.5.

## 1.1 Problem Definition

The aim of this dissertation is to propose various algorithms that infer and exploit contextual visual relations between instances of images. We normally define a relation between a pair of objects as  $\langle (object_1, object_2), relationclass \rangle$  and construct a relation graph structure using the aforementioned relations. Each object usually consists of location and class, and classes of relations can be omitted. Based on the relation definition, each task in this dissertation has its own problem definition to achieve research goals. In this section, we provide detailed problem settings for each task.

The first task is to construct the relation graph between objects in diagrams. Most of the objects in diagrams are presented as illustrations, and the number of object classes is huge. As such, instead of detecting classical object types (such as cats and dogs), we define objects within four categorical classes that are adequate for diagrams: blob (individual object), text, arrow head and arrow tail. Moreover, due to pairwise relations, the relation graph is defined as a bipartite graph. Therefore, this task can be defined as multi-task training that includes an object detector and a graph generator.

The second task is to exploit relation graphs of diagrams to solve the most complex multimodal QA problem, Textbook Question Answering (TQA). TQA datasets include visual and textual contexts, visual and textual questions and

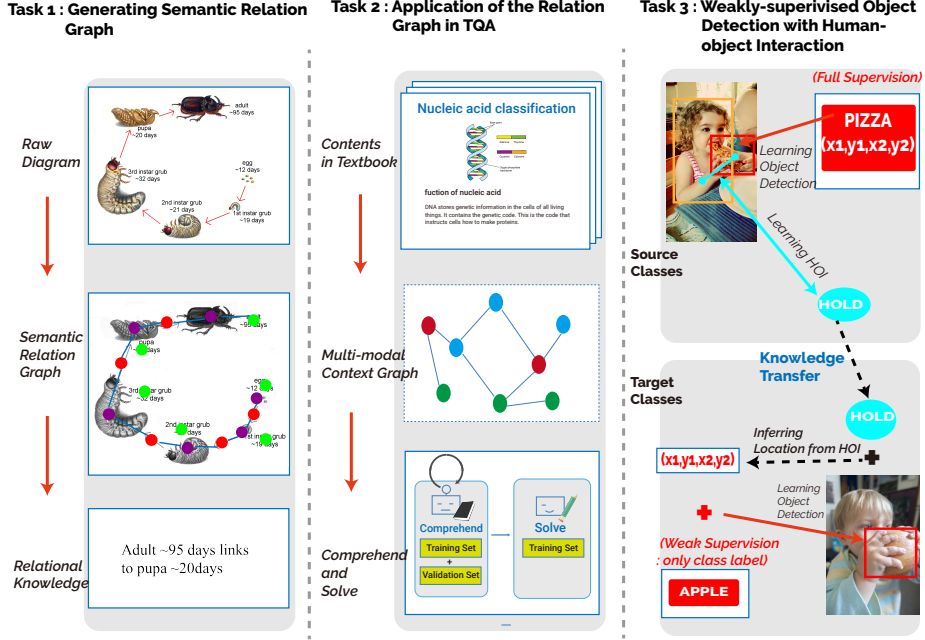


Figure 1.2: Illustration of the tasks addressed in this dissertation. We defined three problems and proposed novel methods for each task: Generating visual Relation Graph from Diagrams, Application of the Relation Graph in Textbook question answering and Weakly-supervised Object Detection with Human-object Interaction

choice candidates. Using all of the visual and textual inputs, the QA model should solve multiple-choice problems in the textbook. Thus, we fuse features from multimodal context graphs by using relation graphs of diagrams and parsing graphs of texts. Moreover, we add a self-supervised pretraining step to learn novel concepts and terms in advance.

Lastly, we introduce new weakly supervised object detection (WSOD) paradigm using visual relation knowledge between humans and objects. In this disserta-



tion, the term “weakly supervised” is used to describe incomplete supervision. In our scheme, we define a source domain as non-rare object classes and a target domain as rare object classes. Without bounding box annotations, our method can train an object detector with only weak verbal supervisions of HOI in the target domain using transferred relation knowledge from the source domain. Therefore, our new model can continuously learn additional object classes with weak supervision annotations.

## 1.2 Motivation

This section discusses the importance of visual relation and the proposed algorithms. The information acquired by visual cognition can be sufficient and crucial to understanding a given situation. When considering various types of information implied in a visual scene, visual relations can play an important role and can be applied to more sophisticated tasks. Simply recognizing an object does not account for contextual information and therefore can hardly be applied to practical applications. For example, in autonomous driving, a car should predict the intentions of surrounding cars and pedestrians rather than just recognizing their locations.

Moreover, displaying visual relations in images can aid in interpreting important evidence about the implicit process of neural networks. While deep learning has contributed to solving problems in various fields, no one can clearly address how a model is working in a “black box”. Therefore, the increased interpretability of algorithms can enable further study of image understanding.

Applying the visual relations in images to practical problems can contribute to making a meaningful baseline in the artificial intelligence field. In particular,

the QA task has been the most promising problem that can be applied in the real world. This dissertation focuses on proposing a novel method to solve the most practical QA task and stimulating the next steps required to develop realistic solutions.

Visual relations as contextual information can also be repeatedly exploited by knowledge transfer. A human can reuse prior knowledge about physical and visual relations between human and object . For example, a human can infer the location of another human’s object if they perceive that the other human is holding something. Therefore, in a visual scene, it can be more efficient to learn about new objects using prior knowledge of semantic relations.

If personal robots are incorporated into homes or offices in the future, we can easily teach them new concepts using accumulated prior knowledge. In order to teach novel information to robots, we can only explain, for example by saying, “That man is holding something. ” We believe that this work is a meaningful step toward achieving general artificial intelligence (GAI).

### **1.3 Challenges**

There are several obstacles to improving the image-understanding ability of machines. In this section, we discuss the challenges for each task.

First, when constructing a visual relation graph, several problems must be addressed. Since a number of objects are arbitrarily located in images, there are inherent difficulties when learning the ensuant relationships. The graph structure can provide advantages to cope with arbitrary layouts. Moreover, since diagrams employ a wide variety of methods in their layout and composition to explain concepts, understanding a diagram requires the challenging task of inferring a

human’s general perception of structured knowledge. Even if pairwise relations are inferred in a graph structure, the correlation among relationships is another challenge . The method used to propagate information between relations can affect results, so the algorithm should be robust for any sequence of inputs.

Second, exploiting visual relations may be negatively impacted by the multimodality of inputs. Humans can easily combine information from multimodal data and reproduce proper information to solve problems. Since the TQA dataset has the highest complexity of data format and length , it is important to extract exact knowledge from long texts and arbitrary images. In addition, various topics and subjects in textbooks are spread over chapters and lessons, and most of the knowledge and terminology do not overlap between chapters and subjects.

Lastly, we tackle the WSOD problem in real-world images using relations between humans and objects. In order to exploit visual relations as context, we should define a relation as a numeric feature that can be integrated into a neural network. Hence, linguistic features can be used to connect a semantic concept with relation features. Moreover, another challenging characteristics for a relation between humans and objects is the ambiguity in a scene. For example, a relationship can be clearly understood with direct contact, but contactless relationships such as watching are not well defined in a scene. Although we can infer a relationship between humans and objects, other valid relationships may also exist. For example, if a baby is holding an apple, the relationship may also be interpreted as eating . Therefore, the proposed algorithm must robustly cope with ambiguity and arbitrary context.

## 1.4 Contributions

The contributions of this dissertation are discussed for each task in this section.

### 1.4.1 Generating Visual Relation Graphs from Diagrams

For generating the visual relation graph, a unified diagram parsing network (UDPnet) is first proposed to understand a diagram by jointly solving the two tasks of object detection and relation matching. An existing study has separately incorporated various traditional algorithms to extract each object class and infer connectivity of nodes. To avoid accumulated errors of the past approach, we simultaneously optimize our unified architecture in an end-to-end manner. At inference, our network has an advantage over simply predicting results in a cascaded manner.

Second, we propose an RNN-based dynamic graph generation network (DGGN) to fully exploit the diagram information by describing the diagram with a graph structure. Since the past work has inferred relations between nodes using only vanilla RNN, it could not fully exploit the graph structure. Regardless of established structure, the vanilla RNN could work in a sequence that may result in the loss of important information. Thus, we propose a dynamic adjacency tensor memory (DATM) in which the DGGN can store information about the relationships between the elements in a diagram. The DATM has features of both an adjacency matrix in graph theory and a dynamic memory in recent deep learning. With this new type of memory, the DGGN offers a novel way to propagate information through the structure of a graph.

In order to demonstrate the effectiveness of the proposed DGGN, we evaluated our model using several diagram datasets. We also analyzed the inside of

GRU cells to observe the dynamics of information in the DGGN. Our model outperformed other baselines and had competitive results for QA datasets.

#### **1.4.2 Application of the Relation Graph in Textbook Question Answering**

First, we suggest a novel architecture with a fusion GCN (f-GCN) to extract knowledge features from the multimodal context graph of long lessons and images in the textbook. Contextual multimodality exists even in non-diagram questions and long textbook lessons must be comprehended to obtain knowledge. Therefore, it is important to extract exact knowledge from long texts and arbitrary images. We establish a multimodal context graph and propose a novel module based on graph convolution networks (GCN) to extract suitable knowledge for solving questions.

Second, we introduce a novel self-supervised learning process to TQA training to comprehend an open-set dataset in order to tackle out-of-domain issues. In textbooks, various topics and subjects are spread over chapters and lessons, and most of the knowledge and terminology do not overlap between chapters and subjects. Therefore, it is very difficult to solve problems for subjects that have not been studied before. To resolve this problem, we encourage our model to learn novel concepts and terms in a self-supervised manner before learning to solve specific questions.

With the proposed model, we were able to obtain state-of-the-art performance on TQA datasets with a large margin compared to current state-of-the-art methods. Moreover, ablation studies validated that both methods of incorporating f-GCN for extracting knowledge from multimodal contexts and our newly proposed self-supervised learning process are effective for TQA problems.

### **1.4.3 Weakly Supervised Object Detection with Human-object Interaction**

First, we define a new weakly supervised object detection scheme that mainly relies on interactions between a human and objects without box annotations. While learning new object classes necessitates considerable effort for manual annotation, context can reinforce supervision without additional effort. Moreover, humans can easily express context with sentences, so linguistic labels can be a key to reduce annotation costs for humans. The objective of this scheme is to make our model learn additional rare classes with weak verbal supervisions that are easily annotated by a human.

Second, we propose a novel module called the “relational region proposal network” (RRPN) that can estimate locations of unseen objects by using human-object interactions. While the RRPN is only trained with weak verbal supervisions of HOI without bounding box annotations in the source domain, it can output an object-localizing attention map in the target domain.

In our experiments, our model demonstrates potential, and our novel scheme effectively leads to meaningful performances compared to the supervised paradigm. We also demonstrate qualitative results that can localize unseen objects in HICO-DET and V-COCO datasets.

## **1.5 Outline**

The structure of this dissertation is as follows: in Chapter 2, background research for three tasks is reviewed. Chapter 3 proposes a method for generating semantic relation graph from diagrams. Chapter 4 presents a solution for Textbook Question Answering. Chapter 5 describes a novel weakly supervised

learning method for object detection that exploits human-object interaction. Finally, Chapter 6 provides concluding remarks, limitations and future directions of this research.

## **Chapter 2**

### **Background**

This chapter provides background information on visual relationships across broad areas. In Section 2.1, prior research about visual relationships is described. Since this dissertation mainly focuses on visual relationships within neural networks, all tasks are related to Section 2.1. To exploit relation graphs in the deep learning area, in Section 2.2 we present research that has examined neural networks in a graph; this research is related to Chapter 4. In Section 2.3, we review previous visual relationship research related to human-object interaction. For natural images, relationships between humans and objects can be one of the most crucial contexts for semantic image understanding related to Chapter 5

#### **2.1 Visual relationships**

Studies on visual relationships have been emerging in the field of computer vision. This line of research includes visual relationship detection (VRD) [43, 37] and scene graph generation (SGG) [27]. Most of these approaches are based on algorithms for grouping elements by relationships, and aiming to find rela-



tionships among the elements. Visual relationship detection addresses the relationships between pairs of objects for which the output is usually formulated as a triplet in the following form: (subject, predicate, object). Recently, Lu *et al.* [43] have presented a dataset that has become a popular benchmark. They have also proposed a new method that consists of a two-step pipeline: first objects are detected, then object pairs are classified based on whether a relationship exists between the image features and a language prior. This approach can be reasonable to exploit semantic characteristics of relationships between objects. Zhang *et al.* [80] have proposed a Visual Translation Embedding network that exploits a low-dimensional space to embed features from the detected objects. The network then predicts the object relationships with a relation translation vector that is designed to use multimodality of deep learning features. Dai *et al.* [11] have addressed a deep relational network using the statistical dependencies between objects and their relationships. Liang *et al.* [39] have recently exploited a reinforcement learning framework related to this problem to traverse a semantic action graph. A directed semantic action graph is built using language priors to provide a representation of semantic relations. Zhuang *et al.* [82] have designed a context-aware interaction classifier that combines the context and the interaction. The classifier they have built is adaptive to context via weights that are context dependent and therefore naturally leads to zero-shot generalizations. Yu *et al.* [76] have used a large amount of external textual data to distill useful knowledge for triplet learning. Peyre *et al.* [53] have focused on weakly supervised relationship learning to transfer visual phrase embeddings from existing training triplets to unseen test triplets using analogies between relations that involve similar objects. The research of Peyre *et al.* [53] has also introduced the UnRel dataset, which is exhaustively annotated for a set of unusual triplets.

Scene graph generation involves building a visually grounded scene graph that consists of the objects and their relationships. Unlike VRD, the scene graph generation aims to build a global scene graph instead of inferring local visual relationship triplets. To jointly generate the scene graph, message passing among the associated objects and predicates is essential in SGG tasks. Most SGG methods regard the context of a node as a valuable cue and apply various propagation mechanisms to communicate information between neighboring nodes over a candidate scene graph. Xu *et al.* [70] have proposed a model that passes messages containing contextual information between a pair of bipartite subgraphs of the scene graph and iteratively refines its predictions using RNNs. This work has provided an important baseline for iterative message passing scheme for further research. Similarly, Li *et al.* [38] have proposed constructing a dynamic graph with object, phrase and caption regions. A feature-refining structure is used to pass messages across the three levels of semantic tasks through the graph. Zellers *et al.* [79] have designed a Stacked Motif Network, which is a new architecture designed to capture higher order motifs in scene graphs. Moreover, by conducting pattern analysis on object labels, this network has introduced strong baselines that outperform prior state-of-the-art models by modeling intra-graph interactions. Yang *et al.* [72] have developed an attentional graph convolutional network that acts on neighboring edges to propagate information between vertices in the candidate scene graph. This work is a novel approach that exploits a graph convolutional network to embed features in the graph structure.

## 2.2 Neural networks on a graph

The notation of graph neural networks was firstly addressed by Gori *et al.* [19]. Micheli [46] and Scarselli *et al.* [59] subsequently developed algorithms to enhance the graph neural networks. To learn a representation of nodes in the graph, these works exploited RNN to propagate related information iteratively to converge on a stable point. However, the computational cost of this method was expensive, and recent studies have proposed strategies to overcome its limitations.

Due to convolutional network advances in the field of computer vision, recent works have adapted the concept of convolution for graph data. Most approaches can be regarded as variants of graph convolutional networks (GCNs). The first primitive research on GCNs has been done by Bruna *et al.* [6], who have proposed a variant of graph convolution based on spectral graph theory. Since that time, there have been several works that have developed spectral-based graph convolutional networks [13, 32, 36]. Kipf *et al.* [32] have proposed the most popular definition of GCN, which simplifies the spectral-based GCN as a first order equation as follows:

$$f(H, \mathcal{A}) = \sigma(\mathcal{A}HW) \quad (2.1)$$

where  $\mathcal{A}$  is an adjacency matrix,  $W$  is learning parameters of a linear layer, and the element-wise operation  $\sigma$  is the activation function.

Another challenge is determining the range of neighborhoods in the graph and developing strategies to propagate information among related nodes. Recently, Niepert *et al.* [49] and Hamilton *et al.* [22] challenged this issue by sampling a fixed-size neighborhood for each node and then performing the aggregation.

## 2.3 Human-object interaction

In various visual relationship problems, visual recognition of human-object interactions (HOI) is crucial to comprehending a scene in a natural image. Early work has studied the mutual context of human poses and objects [75] and Bayesian models [20, 21] with handcrafted features. As a result of the success of deep learning, Chao *et al.* [9] have recently introduced a new large-scale benchmark, “Humans Interacting with Common Objects” (HICO), for HOI recognition, which has been expanded for detection problems in HICO-DET [8]. To solve HICO-DET datasets, various approaches have been proposed. In [8], combined features from human proposals and object regions were used to solve HOI detection. Gkioxari *et al.* [18] have proposed a human and object detector-based approach for estimating a density map based on Faster-RCNN architecture. A recent approach [54] generates the HOI graph and propagates messages between nodes to infer relationships in a parsing graph.

## Chapter 3

### Generating Visual Relation Graphs from Diagrams

Within a decade, performances on classical vision problems such as image classification [23], object detection [17, 41], and segmentation [42] have been largely improved by the use of deep learning frameworks. Based on the great successes of deep learning for such low-level vision problems, a next step could be deriving semantics from images such as relations between objects. For example, to understand a given soccer scene more deeply, it would be very important not only to detect the objects in the image, such as players and a ball but also to figure out the relationships between the objects.

In this chapter, among various vision problems, we aim to understand diagram images, which have played a major role in classical knowledge representation and education. Previously, most machine learning algorithms have focused on extracting knowledge from the information described by natural languages or structured databases (e.g. Freebase [5], Wordnet [47]). In contrast to language-based knowledge, a diagram contains rich illustrations including text, visual information and their relationships, which can depict human's perception of ob-

---

<sup>0</sup>The following chapter was previously published in [31]

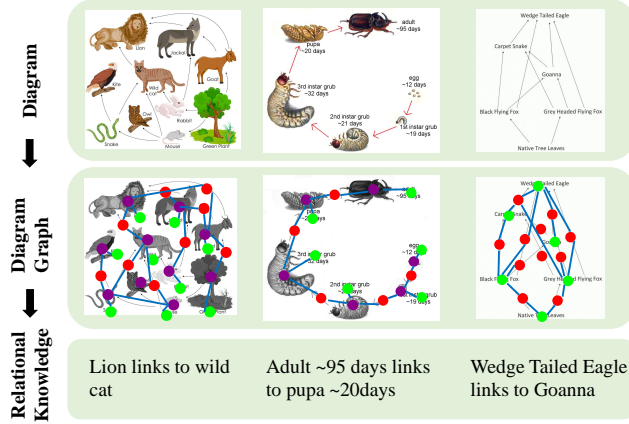


Figure 3.1: Examples of how relational knowledge can be generated from a diagram. In the first row, inputs are only diagrams which have various types of topics, illustrations, texts and layouts. Our model can infer a graphical structure in a diagram as in the second row. In the end, we can extract relational knowledge in the form of sentence from the generated graphs.

jects more succinctly. As shown in Figure 3.1, some complicated concepts such as “food web in a jungle” and “life cycle of a moth” can be easily described as a diagram. On the other hand, a single natural image or a single sentence may not be sufficient to deliver the same amount of information to the readers.

Therefore, we will mainly focus on generating visual relation graphs from diagrams for extracting knowledge described by authors. We propose a novel end-to-end scheme to detect constituents and construct graphs simultaneously. Moreover, an RNN-based module is newly proposed to determine existence of edges among nodes in graphs.

The rest of this chapter is organized as follows: Recent works related to this task is reviewed and several issues of prior work are discussed. We present the

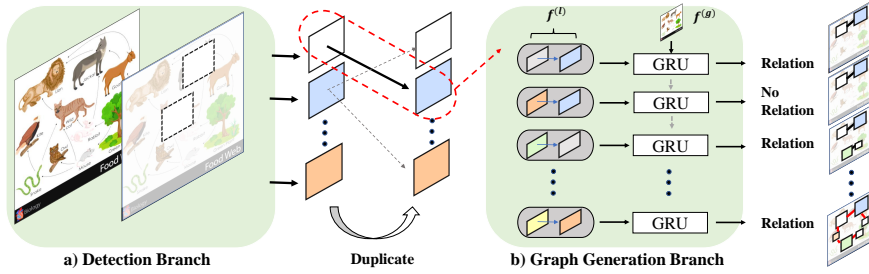


Figure 3.2: Overview of the *unified diagram parsing network* (UDPnet). In (a) the detection branch, an object detector can extract  $n$  objects of 4 different types. Then in order to exploit pairs of objects, we produce  $n^2$  relationship candidates with duplicated objects. (b) In the graph generation branch, we pass local features  $f^{(l)}$  from  $n^2$  candidates to the *dynamic graph generation network* (DGGN) with a global feature  $f^{(g)}$ . In the final step, each relationship candidate can be determined whether it is valid or not. At last, we can establish a relationship graph with nodes and edges.

proposed framework in Section 3.2, and experimental results is demonstrated in Section 3.3. Additional discussions and conclusions of the proposed method is given in Section 3.4 and Section 3.5 respectively.

### 3.1 Related Work

To date there have not been many studies on diagram analysis, but Kembhavi *et al.* [29] have proposed a pioneering work to analyze a diagram’s structure (DS-DPnet). The main flow of the algorithm is twofold: 1) Object detection: Objects in the diagram are detected and segmented individually by conventional methods such as those in [2, 33]. 2) Relation inference: The relationships among

detected objects are inferred by a recurrent neural network (RNN) to transmit contexts sequentially. However, this approach has several limitations. First, concatenating separated methods results in a long pipeline from input to output, which can cause accumulated errors and loss of context in a diagram. Second, and more importantly, the vanilla RNN is not fully capable of dealing with the information formed as a graph structure. In this chapter, we address a novel method to solve the aforementioned issues.

## 3.2 Proposed Method

Figure 3.2 shows a overall framework of the proposed UDPnet. The proposed network consists of two branches: 1) an object detection network, and 2) a graph generation network handling the relations among the detected objects. In the first branch, a set of objects  $O = \{o_i\}_{i=1}^n$  in a diagram image is detected. In the second branch, the relations  $R = \{r_j\}_{j=1}^m$  among the objects are generated. We define an object  $o_i$  as  $\langle location, class \rangle$ , and a relation  $r_j$  in the form of  $\langle o_1, o_2 \rangle$ . Both branches can be optimized simultaneously by a multi-task learning method in an end-to-end manner. After the optimization process, we can use the generated relational information to solve language-based problems such as question-answering.

### 3.2.1 Detecting Constituents in a Diagram

As seen in the Figure 3.1, various kinds of objects can be included in a diagram depending on the information being conveyed. Those objects are usually described in a simplified manner and the number of object classes is huge, which makes detecting and classifying objects more difficult. In our work, instead of



detecting classical object types such as cats and dogs, we define objects in four categorical classes which are adequate for diagrams: blob (individual object), text, arrow head and arrow tail. Due to simplifying problems, the approaches to recognize elements in a diagram are different between ours and the previous work (DSDPnet). As a detector, we used SSD [41] which has been reported to have a robust performance.

### 3.2.2 Generating a Graph of relationships

#### Overall Procedure of Graph Generation

In our method, the relation matching for objects in a diagram is conducted by predicting the presence of an edge between a pair of vertices using graph inference. The nodes and edges of a graph match to the objects and the relations of paired objects, respectively. Therefore, the graph is described as a bipartite graph,

$$G = (V, E), \quad (3.1)$$

where  $V = X \cup Y$  represents the set of paired disjoint vertices  $X \subset V$  and  $Y \subset V$ , and  $E$  denotes edges of the graph each of which connects a pair of nodes  $x \in X$  and  $y \in Y$ . To construct a bipartite graph, we duplicate the detected objects  $O$  as  $O_x$  and  $O_y$  and assume that those two sets are disjoint. Then we predict whether an edge between the nodes  $o_x \in O_x$  and  $o_y \in O_y$  exists.

The connection between nodes is determined by their spatial relationship and the confidence score for each object class which is provided by the object detector. Note that we do not use convolution features from ROI pooling because there can be various kinds of objects in a diagram, whose shape and texture are

hard to be generalized. Instead, we define a feature  $f_x \in \mathbb{R}^{13}$  for the object  $o_x$  including location (xmin, ymin, xmax, ymax), center point (xcenter, ycenter), width, height and confidence scores. Thus, the relationship between two objects  $o_x$  and  $o_y$  is described as local feature  $f^{(l)} = [f_x, f_y] \in \mathbb{R}^{26}$ , and the feature vector  $f^{(l)}$  acts as an input to a RNN layer. To prevent the order of local features in a sequence from affecting the performance, we randomly shuffle the order of the features before training in every iteration.

Furthermore, to extract the layout of a diagram and spatial information of all objects, a global feature  $f^{(g)}$  is utilized as an input to the RNN. The global feature  $f^{(g)} \in \mathbb{R}^{128}$  is constructed by the sum of the convolution feature of conv-7 layer ( $256 \times 1 \times 1$ ) of backbone network in the first branch and the binary mask feature of a diagram ( $128 \times 1$ ). To match the dimension of conv-7 feature as that of hidden units, we use a fully connected layer in the last step. For the mask feature, we pass the  $\mathbb{R}^{n_h \times n_w \times n_c}$  dimensional binary mask map to the 4 layered convolution and max pooling to match the dimension to the hidden unit, where  $n_h$  and  $n_w$  are the width and height of an image, and  $n_c$  is the number of object classes.

### Dynamic Graph Generation Network

In our problem, the local feature vector  $f_{i,j}^{(l)}, (i, j = 1, \dots, n)$  contains the connection information between the nodes  $o_i \in X$  and  $o_j \in Y$ . For simplicity, instead of two indices  $i$  and  $j$ , we will use one index  $t$  to denote the local feature, *i.e.*  $f_t^{(l)}, (t = 1, \dots, n^2)$ . In the previous work [29], vanilla RNN was used and the connection vector  $f_t^{(l)}$  was inputted sequentially to train the RNN. The problem is that there is no guarantee that the input  $f_t^{(l)}$  will be associated with the  $f_{t+1}^{(l)}$  because the vector  $f_t^{(l)}$  is randomly shuffled in stochastic gradient training.

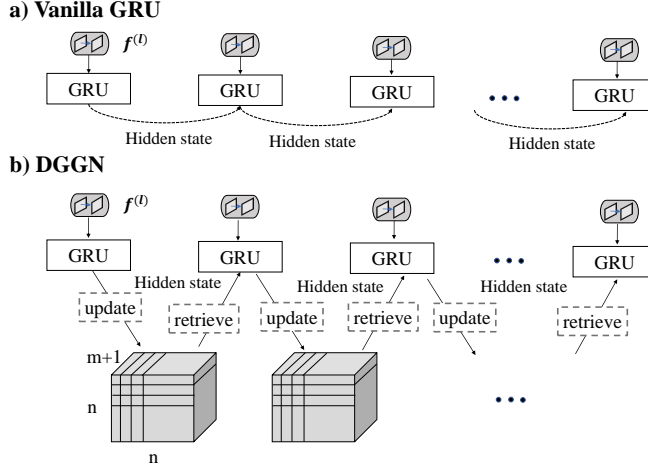
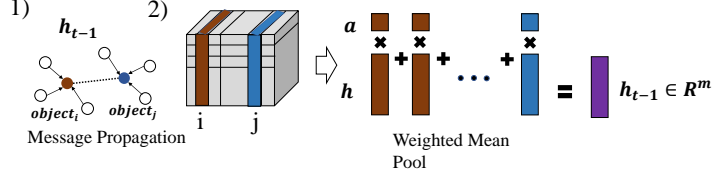


Figure 3.3: Comparison of the vanilla GRU and the proposed DGGN. (a) In vanilla GRU, information is sequentially transmitted only to a randomly selected next cell. (b) In DGGN, past hidden states are calculated with the dynamic adjacency memory, and the information on the entire graph is propagated in both the update and the retrieval processes simultaneously.

Besides, while we define this problem as the bipartite graph inference, vanilla RNN could not capture the graph structure and propagate it into the next unit.

To solve the aforementioned problem, we propose the DGGN method which incorporates GRU as a base model. As shown in Figure 3.3, the proposed method of propagating previous states to the next step is completely different from that of the vanilla GRU. In order to exploit the graph structure, instead of just sequentially transferring features as in vanilla GRU (Figure 3.3(a)), we aggregate messages from adjacent edges (Figure 3.3(b)). To pass the messages from adjacent edges, the proposed DGGN requires a dynamic programming scheme which can build the graph structure in an online manner.

**a) Retrieve step of DGGN**



**b) Update step of DGGN**

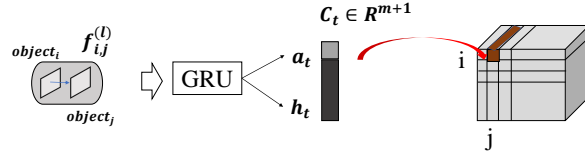


Figure 3.4: Specific explanations of *update* and *retrieve* steps with the DATM in DGGN. (a) In the retrieve step, past messages are transmitted from adjacent edges (a-1). Specifically, to obtain previous hidden state, we conduct weighted mean pool with extracted matrix at indexes of objects. (b) In the update step, model can store the inferred information into the DATM with a concatenated vector at indexes of input objects.

In this chapter, we incorporate the adjacency matrix in the graph theory which has been mainly used to propagate message through known structure of the graph [59]. However, in our problem, the adjacency matrix is unknown, which has to be estimated. Therefore, we propose a dynamic memory component into this problem which holds the connection information among nodes. In this work, we expand 2-dimensional adjacency matrix to 3-dimensional memory. The *dynamic adjacency tensor memory*(DATM)  $D \in \mathbb{R}^{n \times n \times (m+1)}$  is defined as a concatenation of the adjacency matrix  $A \in \mathbb{R}^{n \times n}$  and the corresponding hidden unit  $H$  whose  $(i, j)$  element  $h_{i,j}$  is an  $m$  dimensional hidden vector of the GRU which is related to the connection between the nodes  $o_i$  and  $o_j$ . The

adjacency matrix  $A$  represents the connection status between each of  $n$  nodes in the directed graph. Each cell in the adjacency matrix only indicates whether the corresponding pair of nodes has a directed arc or not. Then both *retrieve* and *update* steps with tensor  $D$  are implemented to aggregate messages from adjacent edges and to build up graph simultaneously.

**Retrieve Step:** Figure 3.4(a) shows the *retrieve step* of DGGN. We can get the previous hidden state  $\hat{h}_{t-1}$  which collects messages propagated through adjacent edges (Figure 3.4 (a-1)). In doing so, as shown in Figure 3.4(a-2) and equation (3.2), we take average of the adjacent vectors of  $o_i$  and  $o_j$  weighted by the probability of the existence of an edge. Formally, we extract an adequate hidden unit  $\hat{h}_t$  for the input vector  $f_{t+1}^{(l)}$  representing the connection with node  $i$  and  $j$ , as in

$$\hat{h}_t = \sum_{k=1}^n a_{k,i} h_{k,i} + \sum_{k=1}^n a_{k,j} h_{k,j} + f^{(g)}. \quad (3.2)$$

Here,  $a_{i,j}$  represents the  $(i, j)$  element of the matrix  $A$ , and  $h_{i,j} \in \mathbb{R}^m$  is the hidden unit stored in the  $(i, j)$  location of the tensor  $H$ . In this step, the probability  $a_{i,j}$  works as weights for aggregating messages which represents the philosophy that more credible adjacent edges should give more credible messages. Before transmitted to GRU layer, the global feature  $f^{(g)}$  is added to reflect the global shape of the diagram.

**Update Step:** In the *update step* shown in Figure 3.4(b), we update the cell  $D_{ij}$  with an  $m + 1$  length vector that concatenates the output  $a_t$  and the hidden state  $h_t$  from a GRU cell (3.8).

$$r_t = \sigma(W_{xr} f_t + W_{hr} \hat{h}_{t-1} + b_r), \quad (3.3)$$

$$z_t = \sigma(W_{xz}f_t + W_{hz}\hat{h}_{t-1} + b_z), \quad (3.4)$$

$$\bar{h}_t = \tanh(W_{xh}f_t + W_{hh}(r_t \odot \hat{h}_{t-1}) + b_h), \quad (3.5)$$

$$h_t = z_t \odot \hat{h}_{t-1} + (1 - z_t) \odot \bar{h}_t, \quad (3.6)$$

$$a_t = \sigma(W_l h_t + b_l), \quad (3.7)$$

$$D_{ij} = [a_t, h_t]. \quad (3.8)$$

Here,  $\sigma(\cdot)$  is a sigmoid function. To obtain the hidden state  $\hat{h}_t$ , the vectors  $\hat{h}_{t-1}$  and  $f_t^{(l)}$  are used as previous hidden state and input vectors of the standard GRU, respectively. Update gate  $z_t$  has a role to adjust influx of previous information  $\hat{h}_{t-1}$  in the GRU cell (3.6). The binary output  $a_t$  is obtained after fully connected layer (3.7).

### 3.2.3 Multi-task Training and Cascaded Inference

In this work, the proposed UDPnet shown in Figure 3.2 is trained in an end-to-end manner. Because the UDPnet consists of two branches (object detection by SSD and graph generation by DGGN), by nature, the problem is a multi-task learning problem. Thus, different losses for each branches are combined into the overall loss  $L$  as follows:

$$L = \alpha L_c + \beta L_l + \gamma L_r. \quad (3.9)$$

The overall loss is a weighted sum of the classification loss  $L_c$  and the location regression loss  $L_l$  for the object detection branch, and the relation classification loss  $L_r$  for the graph generation network.

As defined in original SSD, the classification loss  $L_c$  is the cross-entropy loss over confidences of multiple classes and the location regression loss  $L_l$  is a smooth L1 loss [17] between the predicted box and the ground truth box.

The relation classification loss  $L_r$  is the cross-entropy loss over two classes, adjacent or not. For a faster convergence, we first pre-trained object detection branch alone, then fine-tuned both branches jointly with the overall loss.

During training, matching strategy between the candidates and the ground truths is important for both box detection and relationship inference. To solve the issue, we set our own strategy for matching candidate pairs and the ground truth. First, given  $n$  objects detected at the first branch of object detection, we generate  $n^2$  pairs of relation candidates. For each relation candidate, the two intersection over unions (IOUs), each of which is computed between one of the detected objects and the closest ground truth object, are averaged. Then, each ground truth relationship is matched with the best overlapped relation candidate. To consider the imbalance in the number of detected objects among different diagrams, we should sample the same number of relation candidates from each training diagram.

At inference, we first detect objects in a diagram. Then we apply non maximum suppression (NMS) with an IoU threshold of 0.45 on boxes with scores higher than 0.01. Unlike in training, we should use all boxes that were detected to generate candidate pairs for next branch. Next, we apply graph generation branch to all relation candidates to infer relationship to each other. Finally, we can obtain a diagram graph composed of adjacent edges between nodes with confidence scores higher than 0.1.

After graph inference, we can post-process the generated relational information to further generate knowledge sentences which can be inputs of question answering models. Thus, our methods can make a bridge between visual inference and linguistic reasoning.

### 3.2.4 Details of Post-processing

In this section, we explain a detailed post-processing procedure of the proposed method. Once relationships are determined among objects, we can additionally make new relationship between objects sharing the same intermediate node. For example, given two text objects sharing the same blob object, we can say that one text is linked to another one. Also, given two text objects connected by two intermediate blob objects, we can say equivalently to the previous case. In most case, the text object represents the name or explanation of the connected blob object. Consequently, making further connections by this rule-based algorithm, we can generate sentences using given texts. Using extensively connected texts, we just put an additional phrase of “links to” such as “Lavar links to Fly”. Note that localized text boxes are recognized using Tesseract<sup>1</sup>. Algorithm 1 shows details of post processing.

## 3.3 Experiment

In this section, we validate the performance of the proposed algorithm for the two sub-problems: graph generating and question answering.

### 3.3.1 Datasets.

We performed experiments on two different datasets: AI2D [29] and FOOD-WEBS [34]. AI2D contains approximately 5,000 diagrams representing scientific topics at an elementary school level. Overall, the AI2D dataset contains class-annotation for more than 118K constituents and 53K relationships among

---

<sup>1</sup><https://github.com/tesseract-ocr/tesseract>



---

**Algorithm 1** Post processing algorithm

---

**Require:** Relation set  $R$  generated by the proposed DGGN

**Ensure:** Generated sentences set  $S$

```
1:  $S \leftarrow \emptyset$ 
2: repeat
3:    $R_a \leftarrow \{o_{a1}, o_{a2}\} \in R$ 
4:    $R_b \leftarrow \{o_{b1}, o_{b2}\} \in R$ 
5:   if  $R_a \cap R_b \in \text{'text'}$  then
6:     Continue
7:   else if  $R_a \cap R_b \in \text{'blob'}$  then
8:     if  $R_a - R_b \in \text{'text'}$  &  $R_b - R_a \in \text{'text'}$  then
9:       Generate sentence  $S_{ab}$ 
10:       $S \leftarrow S \cup S_{ab}$ 
11:    else if  $R_a - R_b \in \text{'text'}$  &  $R_b - R_a \in \text{'blob'}$  then
12:      Find  $R_c$  satisfying  $\{R_c \cap R_b \in \text{'blob'}$  &  $R_c - R_b \in \text{'text'}$   $\}$ 
13:      Generate sentence  $S_{ac}$ 
14:       $S \leftarrow S \cup S_{ac}$ 
15:    end if
16:  end if
17: until all elements in  $R$  are visited
```

---

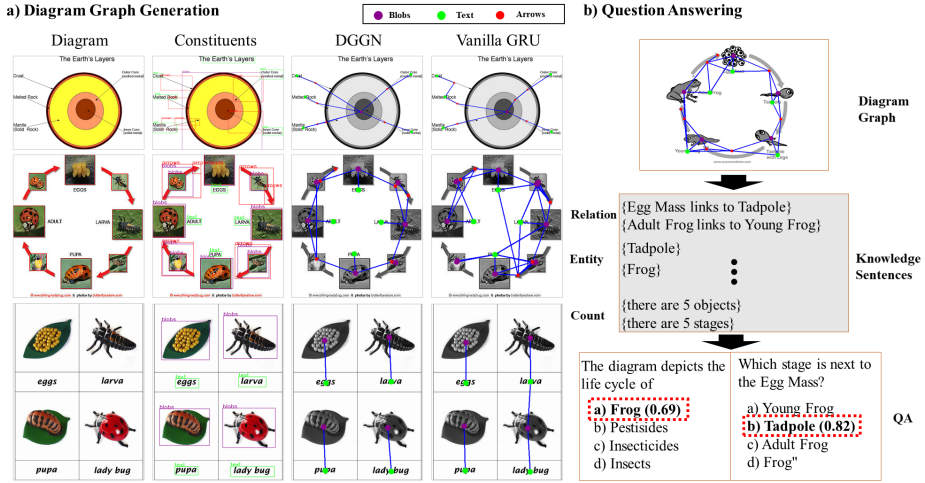


Figure 3.5: Qualitative results of diagram graph generation and a pipeline to solve question answering problem. (a) Each row shows an example of various kinds of diagram. From the left, original diagrams and diagrams with detected constituents are presented. In last two columns, comparison between the DGGN and the Vanilla GRU with final results is shown. (b) From a diagram graph, we extract knowledge sentences, then solve multi-choice problems.

them, including segmentation mask for each of the elements. AI2D also contains more than 15,000 multiple choice questions about diagrams. The polygons for segmentation provided with the AI2D dataset were reshaped into rectangles for simplicity and efficiency. FOODWEBS consists of 490 food web diagrams and 5,208 questions encountered on eighth grade science exams. FOODWEBS focuses on question answering using questions about environmental problems. Unlike AI2D, the diagrams in FOODWEBS do not have annotations for relations among objects, and we used this dataset only as a benchmark of question answering.

### 3.3.2 Baseline.

we used the following ablation models to compare with our method:

- Fully connected layer - only incorporating the object detection branch in our model and replacing the graph generation branch with fully connected layers.
- Vanilla GRU - similar to the previous baseline but using a vanilla GRU instead of the graph generation branch.
- DGGN w/o global feature - exploiting the same structure as our model but excluding the global feature from inputs in the second branch.
- DGGN w/o weighted mean pool - averaging hidden vectors of adjacent edges without multiplying weights which represent the strength of each adjacency.
- DGGN w/ ROI-pooled feature - concatenating a  $2 \times 2$  ROI-pooled feature in the local feature  $f$ , expanding it into a 34 dimensional vector.

### 3.3.3 Metrics.

We propose to measure mean Average Precision (AP) for edge evaluation and IoU for graph completion. First, AP can measure both the recall and precision of a model in predicting the existence of edges. Since our relation candidate should have two boxes, we use average IoUs of those boxes with ground truth boxes as IoU for a relation. We used IoU thresholds  $\tau \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$  for experiments and report the mean AP by averaging the results of all the thresholds.

Additionally, we adopt an IoU metric to measure completion of entire graph, since JIG metric was not clearly defined in the work of DSDPnet. For both of

Table 3.1: Comparison results of AP on the AI2D test set.

Method	$mAP$	$AP_{30}$	$AP_{50}$	$AP_{70}$
Fully connected layer	8.87	9.22	8.92	8.24
Vanilla GRU	39.28	39.89	43.11	31.54
DGGN				
w/o global feature	39.34	40.51	43.03	31.11
w/o weighted mean pool	42.15	44.22	44.99	34.37
w/ ROI-pooled feature	39.73	43.09	42.19	31.38
<b>DGGN</b>	<b>44.08</b>	<b>44.23</b>	<b>47.13</b>	<b>38.97</b>

nodes and edges, we define IoU of node and edge as the number of the intersection divided by the number of the union. Note that we only use the number of overlapped nodes or edges instead of using overlapped area in the original IoU metric.

### 3.3.4 Implementation Details.

For training, We jointly optimized the overall loss of the proposed algorithm with ADAM optimizer with default parameters ( $\beta_2 = 0.999, \epsilon = 10^{-9}$ ). For the three losses in overall loss (3.10), we set  $\alpha = 0.2$ ,  $\beta = 0.1$  and  $\gamma = 1.0$ . The initial learning rate is set to  $1 \times \epsilon^{-4}$  and is multiplied by 0.09 in every 1000 iterations. The batch size is set to 32 and we evaluated our model after 15000 iteration ( $\approx 150$  epochs).

$$L = \alpha L_c + \beta L_l + \gamma L_r. \quad (3.10)$$

We implemented the first branch based on SSDv2 modified from the orig-

inal SSD. For the second branch, we use 1 layer GRU with 128 hidden states. During training, we sample 160 positive and negative relationship candidates at a ratio of 1 to 7. The training and testing codes are built on Pytorch. Additional experiments about QA on diagrams utilized the implementation<sup>2</sup> under the same conditions of previous work (Dqa-Net) [29].

Table 3.2: Comparison results of IoU on the AI2D test set.

Method	$IoU_{node}$	$IoU_{edge}$
Vanilla GRU	70.06	15.58
DGGN		
w/o global feature	<b>70.95</b>	14.44
w/o weighted mean pool	69.48	24.84
w/ ROI-pooled feature	69.24	23.00
<b>DGGN</b>	69.77	<b>25.86</b>

Table 3.3: Comparison results of Recall@K on the AI2D test set.

Method	$R@5$	$R@10$	$R@20$
Vanilla GRU	21.79	33.97	48.87
DGGN			
w/o global feature	21.44	33.63	49.18
w/o weighted mean pool	22.62	35.75	51.60
w/ ROI-pooled feature	21.45	34.21	49.87
<b>DGGN</b>	<b>22.66</b>	<b>35.93</b>	<b>51.73</b>

<sup>2</sup><https://github.com/allenai/dqa-net>

### 3.3.5 Quantitative Results

Table 3.1 shows comparisons DGGN with baselines on the AI2D dataset. Our results demonstrate that the *DGGN* outperforms baselines. In the second row of the table 3.1, the *Fully connected layer* model shows 8.87 mAP, which is extremely low. This is because the relational information among the nodes (elements) is not reflected to fully connected layer. The *vanilla GRU* shows 39.28 mAP, which is lower than those of any variants of *DGGN*. This implies that the vanilla GRU model was not successful for embedding the relational information among the nodes, because the GRU model can only learn the sequential order of the input. In this problem, however, the shuffled order of the relation candidates does not have meaningful sequential knowledge of the relationship.

Next, we performed ablation studies with variants of *DGGN* as presented in the middle of Table 3.1. In the table, we can see that *DGGN w/o global feature* achieved the largest margin to the best model, and this indicates that the global feature can significantly enhance the performance. On the other hand, the result of *DGGN w/o weighted mean pool* is slightly lower than the best model which shows that weights might not be meaningful to the performance. Interestingly, *DGGN w/ ROI-pooled feature* scored a lower mAP in spite of the additional information. One possible reason is that ROI-pooled feature can cause overfit without a sufficient amount of training data, since objects in diagrams are hard to be generalized.

Table 3.2 shows comparisons of the modified *IoU* metric for measuring completion of a graph. In the case of the edge inference, we set 0.5 as the threshold of mean IoU of each predicted box intersecting with a ground truth box and set 0.01 as the threshold of confidence for the adjacency of edges. Since all

models use the same SSD model at the object detection branch, results of the  $IoU_{node}$  are similar to each other. They have slight different performance because of the end-to-end fine-tuning process. For  $IoU_{edge}$ , the *DGGN* shows a better performance than other baselines. Like the results of mAP in Table 3.1, the usage of global feature has a significant impact on the performance.

While we measured AP for evaluation in this paper, we additionally utilized recall metric for measuring retrieval power of relationships due to the sparsity of the relationship. Table 3.3 shows results of Recall@k metric (R@k) on AI2D test dataset. The R@k measures the fraction of ground-truth relationship that appears among the top-k most confident predictions. The results of R@5, R@10 and R@20 demonstrate similar trend of results compared to those of mAPs.

Table 3.4 shows the results of the question answering experiments conducted on AI2D and FOODWEBS. We only compared to previous works in QA accuracy rather than JIG metric due to the difference of detecting methods. For AI2D, we first evaluated Dqa-net with ground truth annotations of diagrams as our upper bound. Our model shows an accuracy of 39.73% which outperforms previous work and approaches upper bound by 2 % margin. On FOODWEBS, we only deploy on trained model with AI2D and extract diagram graphs from entire data. The results show our model demonstrates comparative results. Overall, our model performs better when compared with the VQA [1] method, which estimates the answer directly from a diagram image. These question answering tests reveal a potential for expansion to the linguistic field. Also, this result is meaningful in that our model is not directly designed to solve the QA problem.

Table 3.4: Accuracy of Question Answering on AI2D and FOODWEBS. The results of VQA and DQA-Net(Dsdp) on AI2D and FOODWEBS are refer to [29] and [34], respectively.

Method	AI2D [29]	FOODWEBS [34]
Dqa-Net(GT)	41.55	-
VQA	32.90	56.50
Dqa-Net(Dsdp)	38.47	<b>59.30</b>
<b>Dqa-Net(Ours)</b>	<b>39.73</b>	58.22

### 3.3.6 Qualitative Results

In this section, we analyze qualitative results as shown in Figure 3.5. Three diagrams which have different layouts and topics are presented to compare qualitatively in Figure 3.5(a). For example, diagrams for the same topic “life cycle of a ladybug” in the second and third row have different layouts. Nevertheless, our model can understand different layouts and generate graphs according to the intentions of the diagrams. In the second column, the detection results of the object detection branch, finding four kinds of objects (blob, text, arrow and arrow head) in the diagram, are presented. In the third column, we present the results of graph generation on various diagrams. Then we compare our results to those of the baseline (vanilla GRU) as shown in the last column. As seen in the results, we confirmed that our model correctly connected the links between the objects according to their intended relation, in most case.

Figure 3.5(b) shows a sample describing a pipeline of solving question-answering from a diagram graph. After the diagram of “life cycle of a frog” is converted to a relation graph, we can generate knowledge sentences such as “Adult Frog links to Young Frog” with three categories : “relation”, “entity” and



“count”. Using those sentences, we solved the multi-choice QA problems. For instance, the second question asks for the relationship among the objects in the diagram. We have already generated a knowledge sentence “Egg Mass links to Tadpole”, so the QA model can easily respond “b) tadpole” with a confidence of 0.82. This process can contribute to the solution of various problems related to knowledge of relationships.

In next pages, we present additional qualitative results on diagram graph generation and question answering. We provide results of diagram graph generation of various layouts and topics as depicted from Figure 3.8 to Figure 3.10. The results of DGGN are compared with those of those of vanilla GRU. Ground truths are also shown. In Figure 3.11, we also show pipelines from diagram graph to question answering with post-processing described in the previous section. For two different types of questions, *relationship* and *count*, related sentences are highlighted.

### 3.4 Discussion

In this section, we discuss the effectiveness of DGGN by investigating the GRU cells, and we analyzed the dependency of candidate order of DGGN to compare the results between our model and baseline (vanilla GRU).

**Activation of gates.** To understand the DGGN better, we analyze information dynamics in DGGN. For this, we extracted the activation values of the update gate. In equation (3.6), update gate  $z_t$  obtained from equation (3.4) determines the amount of the received information of the cell from the previous  $\hat{h}_{t-1}$ . By investigating the graph of the update gate’s activation, we can observe that this model meaningfully exploits messages from the past. Obviously, the more up-

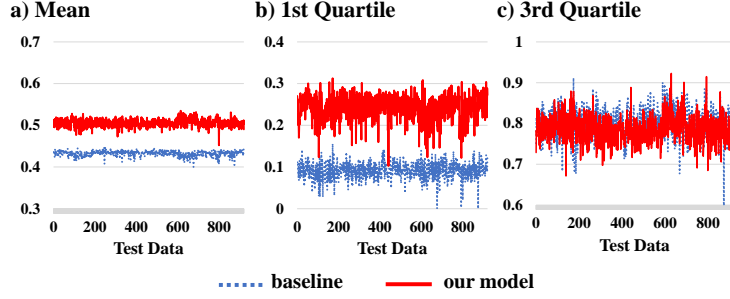


Figure 3.6: Three statistics of activation value of update gate on AI2D test sets. (a) Mean of activation values. (b) The first quartile statistics of activation values. (c) The third quartile statistics of activation values.

date gates activate, the richer the transmitted information becomes.

We plot three statistics of activation values of update gates using 920 test samples. In Figure 3.6(a), we presented the mean of activation values which shows the significant margin between our model (red solid line) and the baseline (blue dotted line), and this shows that our model can generally activate update gates more effectively than the baseline does. While the first quartile statistics in Figure 3.6(b) show a larger margin than the an aforementioned result, the third quartile statistics do not show meaningful differences between our model and the baseline in Figure 3.6(c). Those two results show that our model encouraged activation in relatively inactive update gates. Overall, we can conclude that DGGN delivers more informative messages based on the graphical structure to GRU cells and induces more influx of information from the past which can lead to better results.

For a study in terms of time steps, we extract activation values of update gates in GRU cells from the second diagram in Figure 3.5(a). Then we average

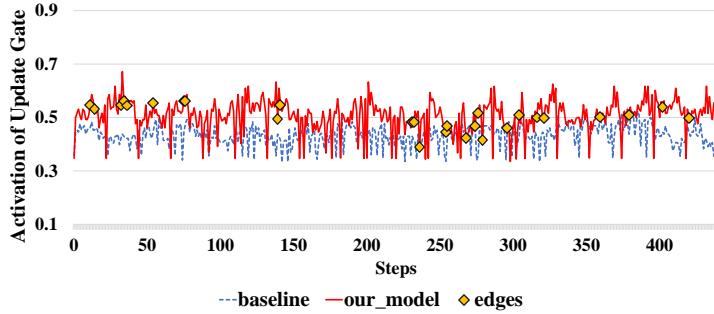


Figure 3.7: Mean of activation values of update gate on second diagram of Figure 3.5.

this quantity over hidden cells with respect to time steps. As shown in Figure 3.7, our model performs higher than the baseline over almost all the steps. Specifically, almost all the yellow dots in the graph, depicting the candidates of the connected edge, show that the activation values of our model are higher than baseline. Therefore, as we discussed in the previous chapter, the cells of our model successfully infer the relationships by accepting more adjacent information with respect to time steps.

**Order of relation candidates.** To explore a mechanism of aggregating messages in DGGN, we verified the effect of the order of relation candidates. We evaluated 50 results ( $AP_{50}$ ) repeatedly with randomly ordered candidates for the baseline and our model on the AI2D. Then we extracted variation statistics from the results. For the baseline, variance and standard deviation of results are  $2.27e^{-5}$  and  $4.76e^{-3}$ , respectively. Our model shows a variance and a standard deviation of results of  $1.03e^{-7}$  and  $3.22e^{-4}$ , respectively. The result shows that the variance and the standard deviation of our model are much lower (around 13

times smaller standard deviation) than those of the baseline.

During the training process, we shuffled the order of candidates before transmitted into GRU cells for both models, to avoid order dependency. However, the statistics show that our model is more robust against the order of relation candidates compared to the baseline. We can confirm that the proposed model successfully extracts the graph structure regardless of the order of the input sequence due to its ability to aggregate messages from the past.

### 3.5 Conclusion

In this work, we proposed *UDPnet* and *DGGN* to tackle the problem of understanding a diagram and generating a graph by the neural network. For diagram understanding, we combine an object detector and a network that generates relations among detected objects. A multi-task learning scheme is used to train the *UDPnet* in an end-to-end manner. Moreover, we propose a novel RNN module to propagate message based on graph structure and generate a graph simultaneously. We demonstrated that the proposed *UDPnet* provides state-of-the-art quantitative and qualitative results on problems of generating relation for a given diagram. We also analyzed how our model works better than strong baselines. Our work can be a meaningful step in diagram understanding and reasoning problem beyond natural image understanding. Moreover, we believe that the *DGGN* could benefit other tasks related to graph structure.

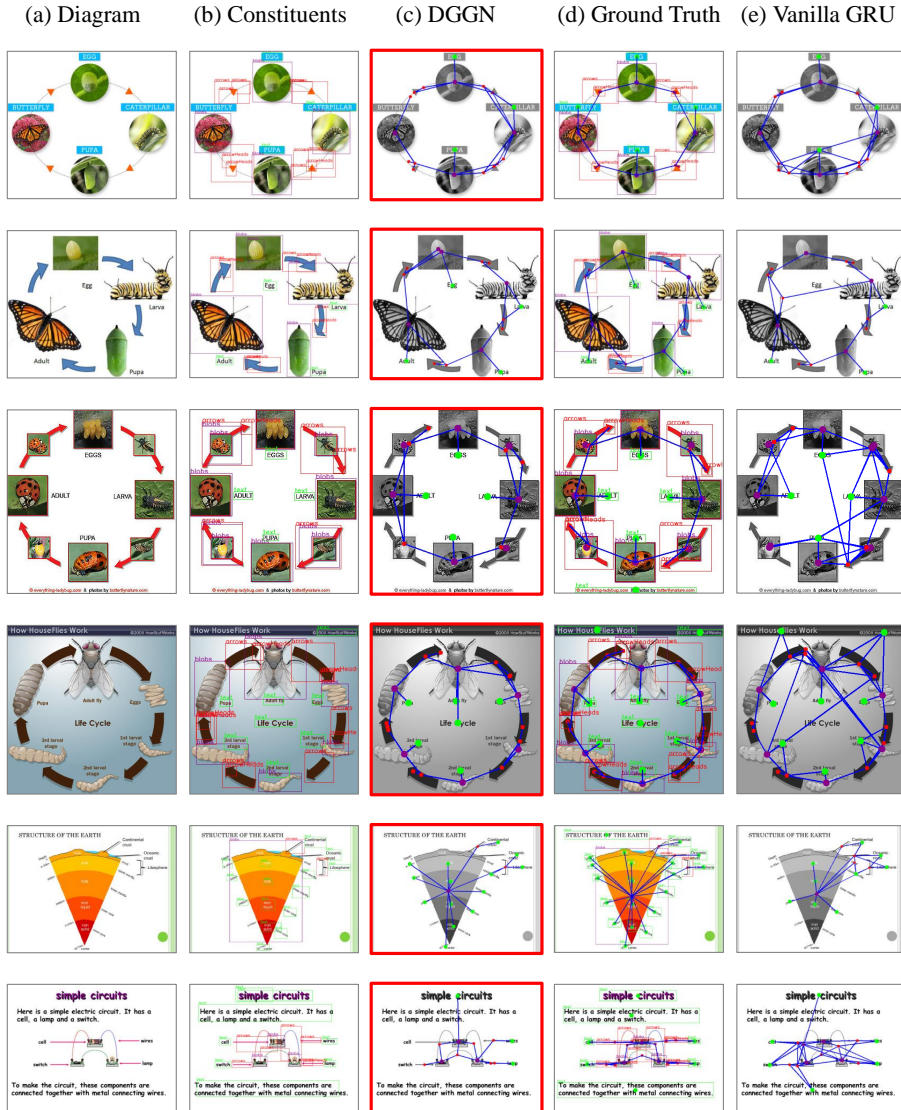
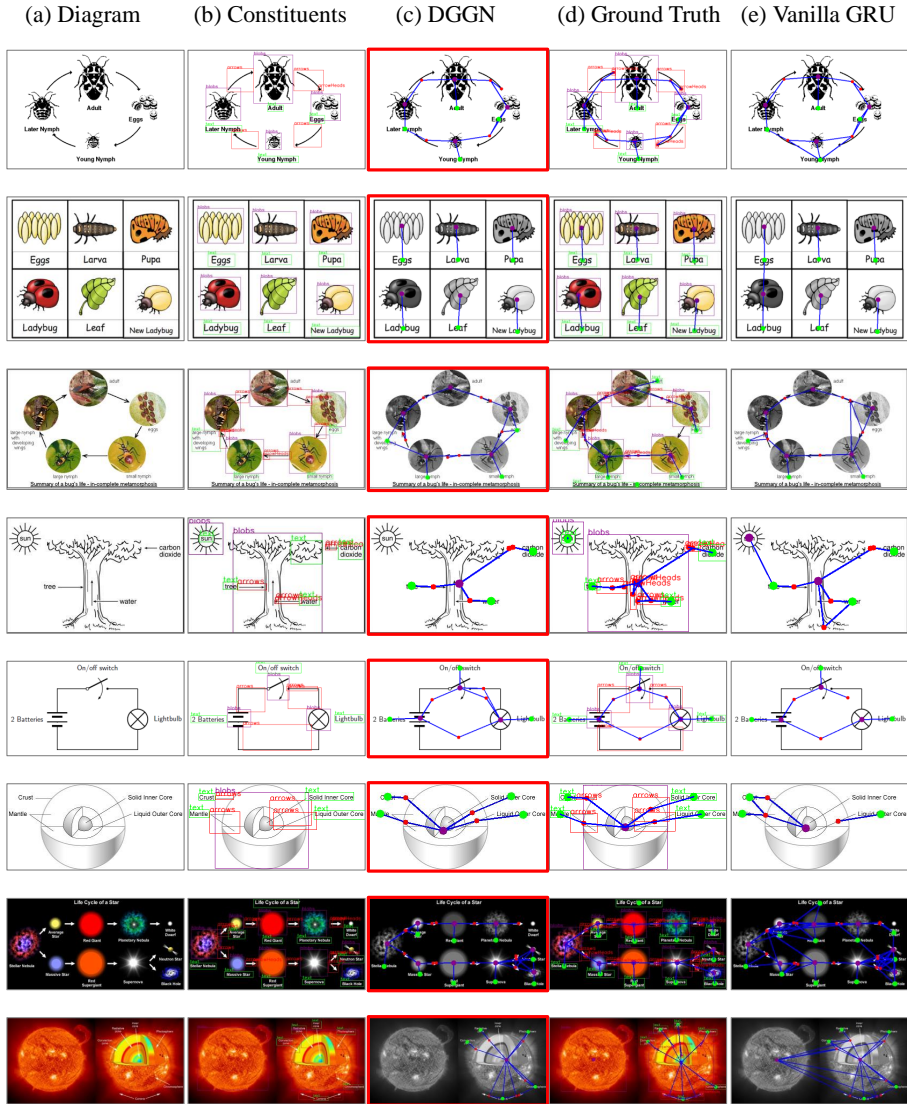


Figure 3.8: Additional qualitative results on diagram graph generation: (a) original diagram (b) diagram with detected constituents (c) generated graph results of DGGN (d) ground truth (e) results of baseline (vanilla GRU)



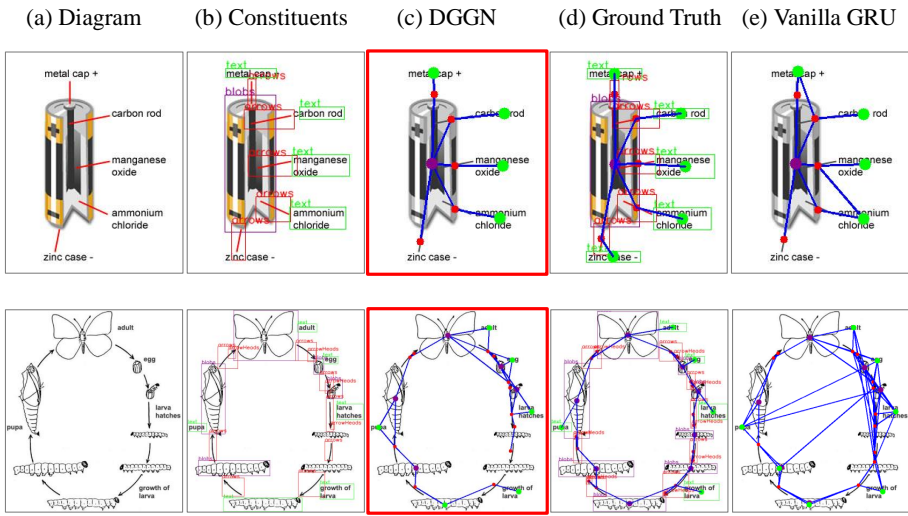


Figure 3.10: Additional qualitative results on diagram graph generation: (a) original diagram (b) diagram with detected constituents (c) generated graph results of DGGN (d) ground truth (e) results of baseline (vanilla GRU)

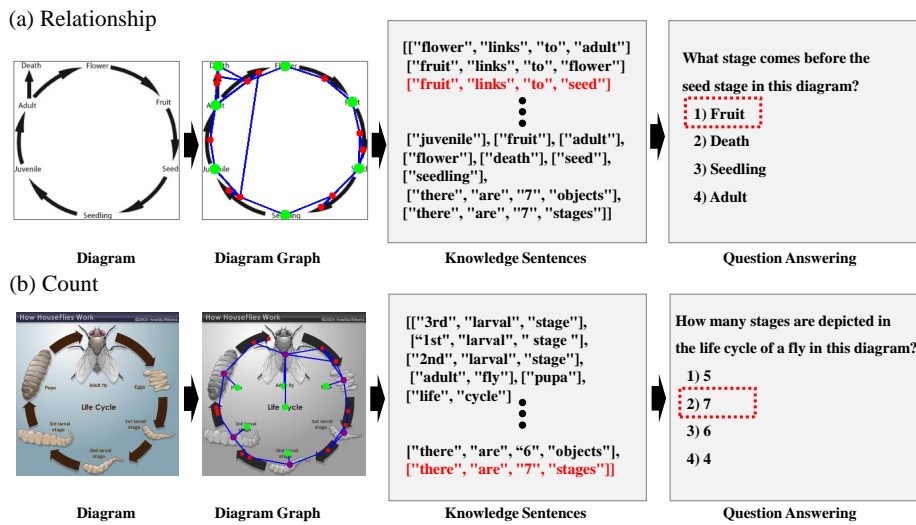


Figure 3.11: Additional qualitative results on question answering: (a) a question about relationship. (b) a question about the count of stages



## **Chapter 4**

### **Application of the Relation Graph in Textbook Question Answering**

In Chapter 3, we present a novel algorithm that establishes relation graphs from diagrams. Since our method extract relational knowledge as graph structure and linguistic sentence, it can be advantageous to apply our solution to a practical problem, Question answering (QA). Moreover, QA has been one of the most promising achievements in the field of natural language processing (NLP) in a decade. Furthermore, it has shown great potential to be applied to real-world problems.

In order to solve more realistic QA problems, input types in datasets have evolved into various combinations. Recently, Visual Question Answering (VQA) has drawn huge attractions as it is in the intersection of vision and language. However, the Textbook Question Answering (TQA) is a more complex and more realistic problem as shown in Table 4.1. Compared to context QA and VQA, the TQA uses both text and image inputs in both the context and the question.

The TQA task can describe the real-life process of a student who learns new

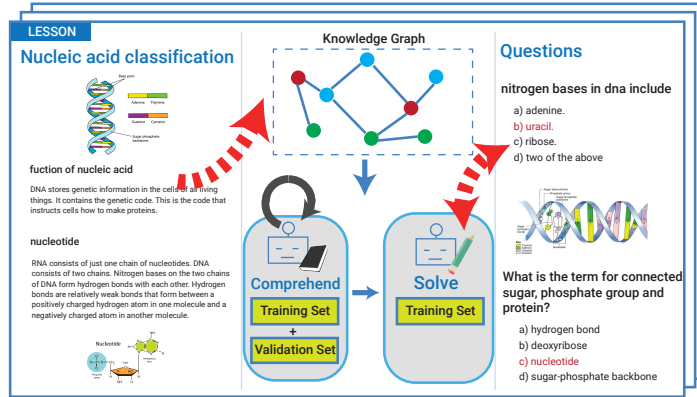


Figure 4.1: Examples of the *textbook question answering* task and a brief concept of our work. In this figure, we can see lessons which contain long essays and diagrams in the TQA. Related questions are also illustrated. With a self-supervised method, our model can comprehend contexts converted into context graphs in training and validation sets. Then it learns to solve questions only in the training set in a supervised manner.

knowledge from books and practices to solve related problems (Figure 4.1). It also has several novel characteristics as a realistic dataset. Since the TQA contains visual contents as well as textual contents, it requires to solve multi-modal QA. Moreover, formats of questions are various which include both text-related questions and diagram-related questions. In this chapter, we address main characteristics of this problem and propose a novel model to tackle aforementioned issues.

The rest of this chapter is organized as follows: We first address recent works related to TQA problem in Section 4.1. We analyze and define the problem in Section 4.2, and we present detailed methods in Section 4.3. Then experimental results is demonstrated and discussed in Section 4.4. Conclusions of the pro-

Table 4.1: Comparison of data types in context and question parts for context QA, VQA and TQA. It shows that the data format of the TQA task is the most complicated on both of context and question parts.

Input Type		Context QA	Visual QA	Textbook QA
Context Part	Text	○	-	○
	Image	-	○	○
Question Part	Text	○	○	○
	Image	-	-	○

posed method is given in Section 4.5.

## 4.1 Related Work

Context question answering, also known as machine reading comprehension, is a challenging task that requires a machine not only to comprehend natural language but also to use reason to answer the asked question correctly. A large number of datasets, such as MCTest [58], SQuAD [55] or MS Marco [48], have contributed significantly to textual reasoning via deep learning approaches; however, these datasets are restricted to a small amount of content and contain just unimodal problems requiring only textual information. In addition, these sets require relatively less complex parsing and reasoning compared to TQA datasets [30]. In this study, we tackle TQA – practical middle school science problems across multiple modalities – by transforming long essays into customized graphs for solving the questions in a textbook.

As the intersection of computer vision, NLP and reasoning, visual question answering has drawn attention in recent years. Most of pioneering work in this

area [71, 73, 44] aims to learn a joint image-question embedding to identify correct answers when the context is proposed by images alone. Various attention algorithms have mainly been developed in this field, and methods of fusing textual and visual information, such as bilinear pooling [16, 77], have also been widely studied. Thereafter, datasets focusing on slightly different purposes have been proposed. For instance, CLEVR [26] has been created to solve the visual grounding problem, and AI2D [29] has suggested a new type of data to extract knowledge from diagrams. In this paper, we incorporate UDPnet [31] to extract knowledge from diagram-parsing graphs in the textbook. Recent research [65, 50] has also dealt with graph structure to solve VQA problems.

While the TQA dataset is newly proposed and has a complicated scheme, several works have been proposed. Kembhavi *et al.* [30] first released the TQA dataset for a competition. This work explained a procedure to produce the dataset and characteristics to be tackled, and the results of the experiments were demonstrated for further studies. The work suggests several baselines that exploit Diagram Parse Graphs [29] to extract features from diagrams. Moreover, a memory network and a machine comprehension model are used to embed texts with an attention mechanism. However, Kembhavi *et al.* [30] have only suggested baselines to validate the potential of the dataset.

Li *et al.* [35] have recommended a new approach to exploit the relation graph structure in this problem. In order to deal with long texts, they have proposed Instructor Guidance with Memory Networks (IGMN), which conducts the TQA task by finding contradictions between the candidate answers and their corresponding context. They have addressed a new discrete structure of Contradiction Entity-Relationship Graphs (CERGs) to represent the facts in the context that may lead to contradictions. However, the approach used predefined rules

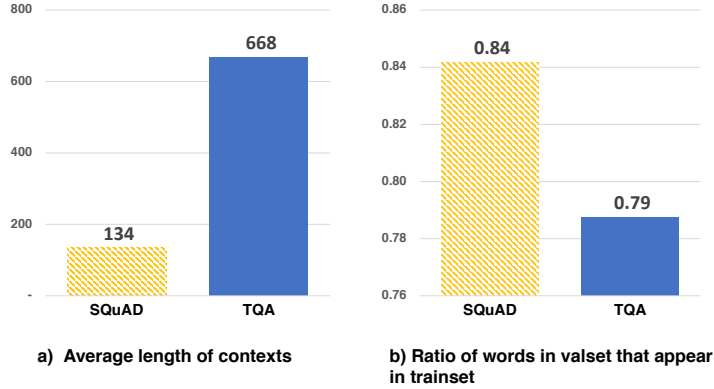


Figure 4.2: Analysis of contexts in TQA and SQuAD datasets.

specific to each category of contradictions that can be conventional in this field, and the attention mechanism was used to jointly exploit the global feature of the multi-modal input, but image features were extracted with a pretrained VGG16 model from images only in contexts. In this chapter, we propose new methods to tackle several limitations of prior work.

## 4.2 Problem

Formally, our problem can be defined as follows:

$$\hat{a} = \operatorname{argmax}_{a \in \Omega_a} p(a|C, q; \theta) \quad (4.1)$$

where  $C$  is given contexts which consist of textual and visual contents and  $q$  is a given question which can contain question diagrams for diagram problems.  $\theta$  denotes the trainable parameters. With given  $C$  and  $q$ , we are to predict the best answer  $\hat{a}$  among a set of possible answers  $\Omega_a$ .

The TQA contexts contain almost all items in textbooks: topic essay, di-

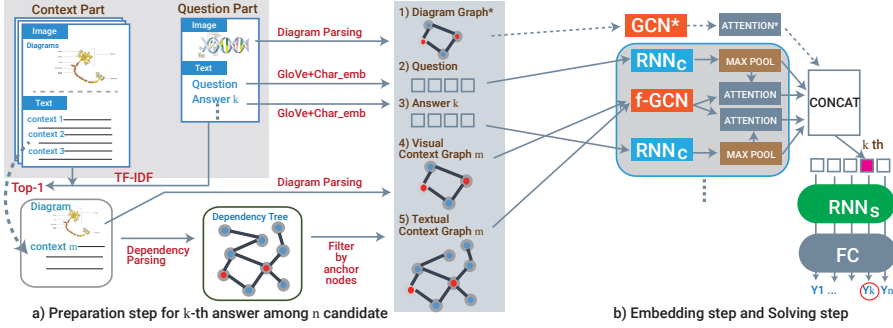


Figure 4.3: **Overall framework of our model:** (a) **The preparation step** for the  $k$ -th answer among  $n$  candidates. The context  $m$  is determined by TF-IDF score with the question and the  $k$ -th answer. Then, the context  $m$  is converted to a context graph  $m$ . The question and the  $k$ -th answer are also embedded by GloVe and character embedding. This step is repeated for  $n$  candidates. (b) **The embedding step** uses  $RNN_C$  as a sequence embedding module and f-GCN as a graph embedding module. With attention methods, we can obtain combined features. After concatenation,  $RNN_S$  and the fully connected module predict final distribution in the solving step.

agrams and images, lesson summaries, vocabularies, and instructional videos. Among them, we mainly use topic essay as textual contexts and diagrams as visual contexts.

Among various issues, the first problem we tackle is the complexity of contexts and variety in data formats as shown in Table 4.1. Especially, analysis of textual context in Figure 4.2(a) shows that the average length of contexts in the TQA is 668 words which is almost 5 times larger than that of the SQuAD which has 134 words on average. Also, in [30], analysis of information scope in TQA dataset provides two important clues that about 80% of text questions only need

1 paragraph and about 80% of diagram questions only need 1 context image and 1 paragraph. Due to those evidences, we need to add an information retrieval step such as TF-IDF (term frequency–inverse document frequency) to narrow down scope of contexts from a lesson to a paragraph, which significantly reduces the complexity of a problem. Moreover, a graph structure can be suitable to represent logical relations between scientific terms and to merge them with visual contexts from diagrams. As a result, we decide to build a multi-modal context graph and obtain knowledge features from it.

In Figure 4.2(b), we obtain the percentage of how much the terms in the validation set are appearing in the training set. Obviously, the ratio of the TQA (79%) is lower than that of the SQuAD (84%) which can induce out-of-vocabulary and domain problems more seriously in the TQA task. To avoid aforementioned issues, we apply a novel self-supervised learning process before learning to solve questions.

We denote the question text, question diagram, candidate answer, text context and diagram context as  $Q^t = \{q_1^t, q_2^t, \dots, q_I^t\}$ ,  $Q^d = \{q_1^d, q_2^d, \dots, q_J^d\}$ ,  $A = \{a_1, a_2, \dots, a_K\}$ ,  $C^t = \{c_1^t, c_2^t, \dots, c_L^t\}$ , and  $C^d = \{c_1^d, c_2^d, \dots, c_M^d\}$ , respectively where  $q_i^t/q_j^d/a_k/c_l^t/c_m^d$  is the  $i^{th}/j^{th}/k^{th}/l^{th}/m^{th}$  word of the question text  $Q^t$  and the question diagram  $Q^d$ , candidate answer  $A$ , text context  $C^t$  and diagram context  $C^d$  ( $C$  is unified notation for the  $C^t$  and  $C^d$ ). The corresponding representations are denoted as  $h_q^t, h_q^d, h_a, H_c^t$  and  $H_c^d$ , respectively. Note that we use the diagram context  $C^d$  only in the diagram questions.

## 4.3 Proposed Method

Figure 4.3 illustrates our overall framework which consists of three steps. In a preparation step, we use TF-IDF to select the paragraph most relevant to the given question or candidate answers. Then, we convert it into two types of context graphs for text and image, respectively. In the embedding step, we exploit an RNN (denoted as  $RNN_C$  in the figure) to embed textual inputs, a question and an answer candidate. Then, we incorporate f-GCN to extract graph features from both the visual and the textual context graphs. After repeating previous steps for each answer candidate, we can stack each of concatenated features from the embedding step. We exploit another RNN ( $RNN_S$ ) to cope with the variable number of answer candidates which varies from 2 to 7 that can have sequential relations such as “none of the above” or “all of the above” in the last choice. Final fully connected layers decide probabilities of answer candidates. Note that notation policies are included in the supplementary.

### 4.3.1 Multi-modal Context Graph Understanding

#### Visual and Textual Context graphs

For the visual contexts and the question diagrams, we build a visual context graph using UDPnet [31]. We obtain names, counts, and relations of entities in diagrams. Then we can establish edges between related entities. Only for question diagrams, we use counts of entities transformed in the form of a sentence such as “There are 5 objects” or “There are 6 stages”.

We build the textual context graphs using some parts of the lesson where the questions can focus on solving problems as follows. Each lesson can be divided into multiple paragraphs and we extract one paragraph which has the highest



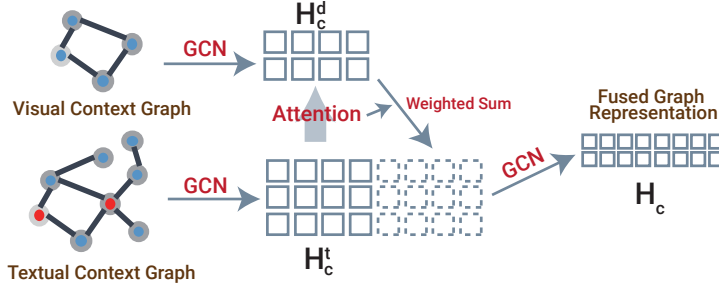


Figure 4.4: Illustration of f-GCN. Both of textual and visual contexts are converted into  $H_c^d$  and  $H_c^t$ . With attention methods, we obtain combined features of  $H_c^t$  and  $H_c^d$  (f-GCN1). Finally, we use GCN again to propagate over entire features of context graphs (f-GCN2).

TF-IDF score using a concatenation of the question and one of the candidate answers (leftmost of Figure 4.3(a)).

Then, we build the dependency trees of the extracted paragraph utilizing the Stanford dependency parser [45], and designate the words which exist in the question and the candidate answer as anchor nodes. The nodes which have more than two levels of depth difference with anchor nodes are removed and we build the textual context graphs using the remaining nodes and edges (Process 1 in the supplementary).

### Graph Understanding using f-GCN

Next, we propose f-GCN to extract combined graph features for visual and textual context graphs as shown in Figure 4.4. Each of context graphs has its own graph matrix  $C$  containing node features and a normalized adjacency matrix which are used as inputs of a GCN to comprehend the contexts. Here, the graph

matrix  $C$  is composed of the word embeddings and the character representation. First, we extract propagated graph features from both of context graphs based on one-layer GCN as

$$\begin{aligned} H_c^t &= f(C^t, \mathcal{A}^t) = \sigma(\mathcal{A}^t C^t W^t) \\ H_c^d &= f(C^d, \mathcal{A}^d) = \sigma(\mathcal{A}^d C^d W^d), \end{aligned} \quad (4.2)$$

where  $\mathcal{A}^t$  and  $\mathcal{A}^d$  are the adjacency matrices for the text and visual contexts,  $W^t$  and  $W^d$  are learning parameters of linear layer for the text and visual contexts, and the element-wise operation  $\sigma$  is the tanh activation function.

After that, we use dot product function to get attention matrix  $Z$  of visual context  $H_c^d$  against textual context  $H_c^t$  which contains main knowledge. Then we concatenate features of textual context  $H_c^t$  and weighted sum  $Z^T H_c^d$  to get entire context features,

$$H_c^1 = [H_c^t; Z^T H_c^d], \quad (4.3)$$

where  $[\cdot; \cdot]$  is the concatenation operator. Compared to the textual-context-only case, we can obtain double-sized features which can be more informative. Finally, we use a GCN again to propagate over entire features of context graphs:

$$H_c^2 = f(H_c^1, \mathcal{A}^t) = \sigma(\mathcal{A}^t H_c^1 W^c). \quad (4.4)$$

We denote this module except the last GCN as f-GCN1 (eq. (4.3)) and the whole module including the last GCN as f-GCN2 (eq. (4.4)).

### 4.3.2 Multi-modal Problem Solving

The f-GCN and RNNs are used to embed the contexts and answer the questions as shown in Figure 4.3(b). Two different RNNs are used in our architecture. One is the *comprehending* RNN ( $\text{RNN}_C$ ) which can understand questions and

candidate answers and the other is the *solving* RNN ( $RNN_S$ ) which can answer the questions.

The input of the  $RNN_C$  is comprised of the word embedding, character representation and the occurrence flag for both questions and candidate answers. In word embedding, each word can be represented as  $e_{q_i}/e_{a_i}$  by using a pre-trained word embedding method such as GloVe [52]. The character representation  $c_{q_i}/c_{a_i}$  is calculated by feeding randomly initialized character embeddings into a CNN with the max-pooling operation. The occurrence flag  $f_{q_i}/f_{a_i}$  indicates whether the word occurs in the contexts or not. Our final input representation  $q_i^w$  for the question word  $q_i$  in  $RNN_C$  is composed of three components as follows:

$$\begin{aligned} e_{q_i} &= Emb(q_i), \quad c_{q_i} = Char-CNN(q_i) \\ q_i^w &= [e_{q_i}; c_{q_i}; f_{q_i}]. \end{aligned} \quad (4.5)$$

The input representation for the candidate answers is also obtained in the same way as the one for the question. Here,  $Emb$  is the trainable word embeddings and  $Char-CNN$  is the character-level convolutional network. To extract proper representations for the questions and candidate answers, we apply the step-wise max-pooling operation over the  $RNN_C$  hidden features.

Given each of the question and the candidate answer representations, we use an attention mechanism to focus on the relevant parts of the contexts for solving the problem correctly. The attentive information  $Att_q$  of the question representation  $h_q$  against the context features  $H_c$  as in (4.3) or (4.4) is calculated as follows:

$$\begin{aligned} Att_q &= \sum_{k=1}^K \alpha_k H_{c_k}, \quad \alpha_k = \frac{\exp(g_k)}{\sum_{i=1}^K \exp(g_i)}, \\ g_k &= h_q^T \mathbf{M} H_{c_k}. \end{aligned} \quad (4.6)$$

Here,  $K$  is the number of words in the context  $C$  which equals the dimension of the square adjacency matrix  $\mathcal{A}$ .  $\mathbf{M}$  is the attention matrix that converts the question into the context space. The attentive information of the candidate answers  $Att_a$  is calculated similar to  $Att_q$ .

$RNN_S$  can solve the problems and its input consists of the representations of the question and the candidate answer with their attentive information on the contexts as:

$$\begin{aligned} I_{RNN_S}^t &= [h_q; h_a; Att_q^c; Att_a^c], \\ I_{RNN_S}^d &= [h_q; h_a; Att_q^c; Att_a^c; Att_q^{qd}; Att_a^{qd}] \end{aligned} \tag{4.7}$$

where  $I_{RNN_S}^t$  is for the text questions and  $I_{RNN_S}^d$  is for the diagram questions. Finally, based on the outputs of  $RNN_S$ , we use one fully-connected layer followed by a softmax function to obtain a probability distribution of each candidate answer and optimize those with cross-entropy loss.

### 4.3.3 Self-supervised open-set comprehension

To comprehend out-of-domain contexts, we propose a self-supervised prior learning method as shown in Figure 4.5. While we exploit the same architecture described in the previous section, we have reversed the role of the candidate answer and the contexts in (4.1) as a self-supervised one. In other words, we set the problem as inferring the Top-1 context for the chosen answer candidate. We assume TF-IDF to be quite reliable in measuring closeness between texts.

The newly defined self-supervised problem can be formalized as follows:

$$\hat{c} = \operatorname{argmax}_{c \in \Omega_c} p(c | A_k, q; \theta) \tag{4.8}$$

where  $A_k$  is given  $k$ -th answer candidate among  $n$  candidates and  $q$  is the given

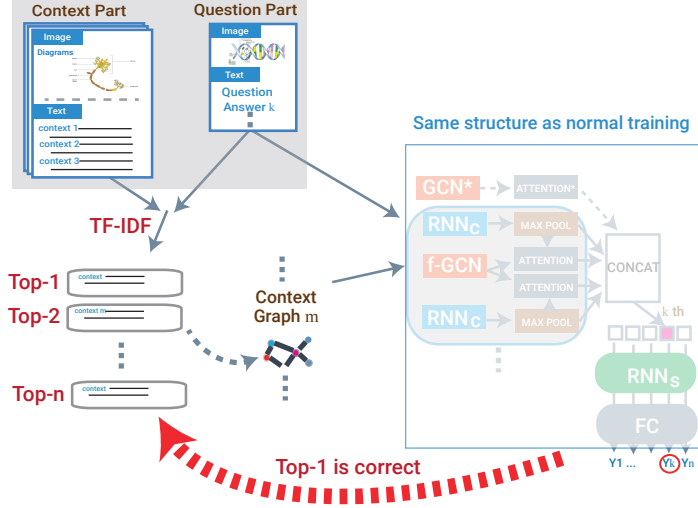


Figure 4.5: Self-supervised open-set comprehension step in our model. We set contexts as candidates we should predict for the question and the  $k$ -th answer. For each answer, we obtain  $n$  context candidates from TF-IDF methods and set the top-1 candidate as the correct context. While we use the same structure as in Figure 4.3, we can predict final distribution after all the steps.

question. Then we infer the most related context  $\hat{c}$  among a set of contexts  $\Omega_c$  in a lesson.

For each candidate answer  $A_k (k = 1, \dots, n)$ , we get the set of paragraphs  $\Omega_c$  of size  $j$  from the corresponding context. Here,  $\Omega_c$  is obtained by calculating TF-IDF between  $[q; A_k]$  and each paragraph  $\omega$ , i.e.,  $T_\omega = tf-idf([q; A_k], \omega)$ , and selecting the top- $j$  paragraphs. Among the  $j$  paragraphs  $\omega_i (i = 1, \dots, j)$  in  $\Omega_c$ , the one with the highest TF-IDF score is set as the ground truth:

$$y_i = \begin{cases} 1, & \text{if } \omega_i = \operatorname{argmax}_{\omega \in \Omega_c} T_\omega, \\ 0, & \text{otherwise.} \end{cases} \quad (4.9)$$

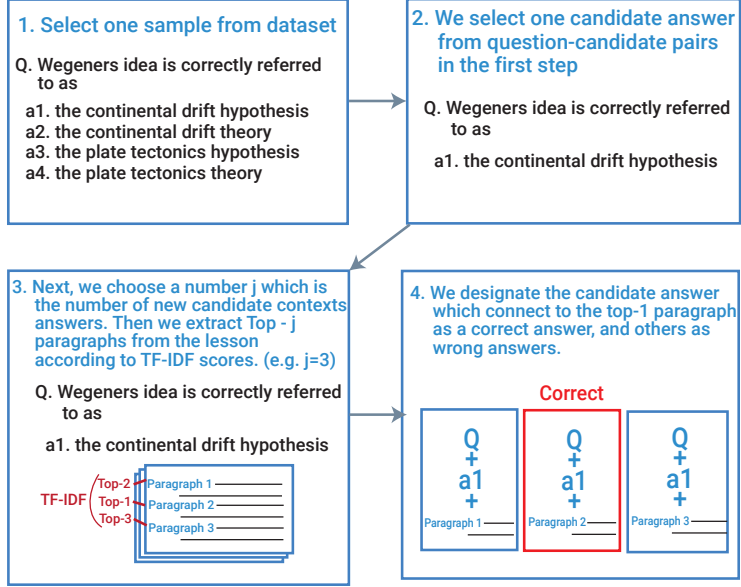


Figure 4.6: Additional examples of SSOC steps.

With  $A_k$ ,  $q$  and  $\omega_i \in \Omega_c$ , we conduct the same process in eq. (2-7) to obtain the  $i$ -th input of the  $RNN_S$ ,  $I_{RNN_S}^i$ . After repeating it  $j$  times, we put all  $I_{RNN_S}^i$ , ( $i = 1 \cdots j$ ) into  $RNN_S$  sequentially and optimize this step with the cross-entropy loss. We repeatedly choose all answer candidates  $A_k$ , and conduct the same process in this step.

In Figure 4.6, we illustrate examples about detailed steps of SSOC. In the first step, we select one candidate answer from question-candidate answers pairs (2). Next, we choose a number  $j$ , the number of candidate contexts for the pair of question-candidate answer, in the range 2 to 7 like the original dataset (3). If  $j$  is higher than the number of contexts in the lesson, we set  $j$  to be the number of contexts. Then, we extract top  $j$  paragraphs using the TF-IDF scores to set them as candidate contexts  $\Omega_c$  (3). We build each context graph in the same way

Table 4.2: Comparison of performance with previous methods (Top) and results of ablation studies (Bottom). We demonstrate the accuracies of each type of questions, Text T/F (true-false in text only), Text MC (multiple-choices in text only), Text all (all in text only), Diagram and All. Note that previous methods only used textual context.

Model	Text T/F	Text MC	Text All	Diagram	All
Random	50.10	22.88	33.62	24.96	29.08
MemN+VQA [30]	50.50	31.05	38.73	31.82	35.11
MemN+DPG [30]	50.50	30.98	38.69	32.83	35.62
BiDAF+DPG [30]	50.40	30.46	38.33	32.72	35.39
Challenge	-	-	45.57	35.85	40.48
IGMN [35]	57.41	40.00	46.88	36.35	41.36
Our full model w/o visual context	62.32	49.15	54.35	36.61	45.06
Our full model w/ f-GCN2	62.22	48.76	54.11	<b>37.72</b>	45.52
Our full model	<b>62.73</b>	<b>49.54</b>	<b>54.75</b>	37.61	<b>45.77</b>
w/o SSOC(VAL)	62.22	48.82	54.11	37.47	45.39
w/o SSOC(TR+VAL)	60.02	46.86	52.06	36.61	43.97
w/o f-GCN & SSOC(TR+VAL)	58.72	45.16	50.51	35.67	42.74

as the original method and get embeddings with the question-candidate answer pair we selected. Finally, we designate the final candidate which connects to the top 1 paragraph as a correct answer, and others as wrong answers (4).

With this pre-training stage which shares parameters with the supervised stage, we expect that our model can deal with almost all contexts in a lesson. Moreover, it becomes possible to learn contexts in the validation set or the test set with a self-supervised manner. This step is analogous to a student who reads and understands a textbook and problems in advance.

#### 4.3.4 Process of Building Textual Context Graph

---

**Process 2** Build textual context and adjacency matrices  $C, \mathcal{A}$

---

**Input:** a paragraph, a set of *anchor nodes*  $V$

- 1: Construct a dependency tree on each sentence of the given paragraph
- 2: Split the tree into multiple units each of which represents two nodes and one edge  $u = \{v_1, v_2\}$
- 3:  $U \leftarrow$  a set of units
- 4:  $E \leftarrow$  an empty set of edges
- 5: **for**  $depth \leftarrow 1$  to 2 **do**
- 6:     **for** all nodes  $v \in V$  **do**
- 7:         **for** all units  $u \in U$  **do**
- 8:             **if**  $v \in u$  **then**
- 9:                  $E \leftarrow E \cup \{u\}$
- 10:             **end if**
- 11:         **end for**
- 12:     **end for**
- 13:      $V \leftarrow$  a set of all nodes in  $E$
- 14: **end for**

**Output:** context matrix  $C$  from  $V$  with embedding matrices, adjacency matrix  $\mathcal{A}$  from  $E$

---

The procedure for converting the textual context into the graph structures is shown in Process 2. After constructing the dependency trees, we set the nodes included in the question or the candidate answer as anchor nodes and built the final context graph  $C$  by removing the nodes which have more than two levels of depth difference with anchor nodes. We also constructed the adjacency matrix



$\mathcal{A}$  using the remaining nodes and edges.

## 4.4 Experiment

### 4.4.1 Implementation Details

We initialized word embedding with 300d GloVe vectors pre-trained from the 840B Common Crawl corpus, while the word embeddings for the out-of-vocabulary words were initialized randomly. We also randomly initialized character embedding with a 16d vector and extracted 32d character representation with a 1D convolutional network. And the 1D convolution kernel size is 5. We used 200 hidden units of Bi-LSTM for the  $\text{RNN}_c$  whose weights are shared between the question and the candidate answers. The maximum sequence length of them is set to 30. Likewise, the number of hidden units of the  $\text{RNN}_s$  is the same as the  $\text{RNN}_c$  and the maximum sequence length is 7 which is the same as the number of the maximum candidate answers. We employed 200d one layer GCN for all types of graphs, and the number of maximum nodes is 75 for the textual context graph, 35 for the diagrammatic context graph, and 25 for the diagrammatic question graph, respectively. We use tanh for the activation function of the GCN. The dropout was applied after all of the word embeddings with a keep rate of 0.5. The Adam optimizer with an initial learning rate of 0.001 was applied, and the learning rate was decreased by a factor of 0.9 after each epoch.

### 4.4.2 Dataset

We perform experiments on the TQA dataset, which consists of 1,076 lessons from Life Science, Earth Science and Physical Science textbooks. While the dataset contains 78,338 sentences and 3,455 images including diagrams, it also

has 26,260 questions with 12,567 of them having an accompanying diagram, split into training, validation and test at a lesson level. The training set consists of 666 lessons and 15,154 questions, the validation set consists of 200 lessons and 5,309 questions and the test set consists of 210 lessons and 5,797 questions. Since evaluation for test is hidden, we only use the validation set to evaluate our methods.

#### 4.4.3 Baselines

We compare our method with several recent methods as followings:

- **MemN+VQA, MemN+DPG** Both exploits Memory networks to embed texts in lessons and questions. First method uses VQA approaches for diagram questions, and the second one exploits Diagram Parse Graph (DPG) as context graph on diagrams built by DsDP-net [29].
- **BiDAF+DPG** It incorporates BiDAF (Bi-directional Attention Flow Network) [60], a recent machine comprehension model which exploits a bi-directional attention mechanism to capture dependencies between question and corresponding context paragraph.

For above 3 models, we use experimental results newly reported in [35].

- **Challenge** This is the one that obtained the top results in TQA competition [30]. The results in the table are mixed with each of top score in the text-question track and the diagram-question track.
- **IGMN** It uses the Instructor Guidance with Memory Nets (IGMN) based on Contradiction Entity-Relationship Graph (CERG). For diagram questions, it only recognizes texts in diagrams.

- **Our full model w/o visual context** This method excludes visual context to compare with previous methods on the same condition. It uses only one-layer GCN for textual context and self-supervised open-set comprehension (SSOC).
- **Our full model w/ f-GCN2** From now, all methods include visual context. This method uses f-GCN2 and SSOC.

Following methods are for our ablation study:

- **Our full model** This method uses both of our methods, f-GCN1 and SSOC on the training and the validation sets.
- **Our model w/o SSOC (VAL)** This method only uses training set to pre-train parameters in SSOC.
- **Our model w/o SSOC (TR+VAL)** This method eliminates whole SSOC pre-training process. It only uses f-GCN as Graph extractor and was trained only in a normal supervised learning manner.
- **Our model w/o f-GCN & SSOC (TR+VAL)** This method ablates both f-GCN module and SSOC process. It replaces f-GCN as vanilla RNN, other conditions are the same.

#### 4.4.4 Quantitative Results

##### Comparison of Results

Overall results on TQA dataset are shown in Table 4.2. The results show that all variants of our model outperform other recent models in all type of question. Our best model shows about 4% higher than state-of-the-art model in overall accuracy. Especially, an accuracy in text question significantly outperforms other

results with about 8% margin. A result on diagram questions also shows more than 1% increase over the previous best model. We believe that our two novel proposals, context graph understanding and self-supervised open-set comprehension work well on this problem since our models achieve significant margins compared to recent researches.

Even though our model w/o visual context only uses one-layer GCN for textual context, it shows better result compared to MemN+VQA and MemN+DPG with a large margin and IGMN with about 3% margin. IGMN also exploits a graph module of contraction, but ours outperforms especially in both text problems, T/F and MC with over 5% margin. We believe that the graph in our method can directly represents the feature of context and the GCN also plays an important role in extracting the features of our graph.

Our models with multi-modal contexts show significantly better results on both text and diagram questions. Especially, results of diagram question outperform over 1% rather than our model w/o visual context. Those results indicate that f-GCN sufficiently exploits visual contexts to solve diagram questions.

### **Ablation Study**

We perform ablation experiments in Table 4.2. Our full model w/ f-GCN2 can achieve best score on diagram questions but slightly lower scores on text questions. Since the overall result of our full model records the best, we conduct ablation study of each module of it.

First, we observe an apparent decrease in our model when any part of modules is eliminated. It is surprising that self-supervised open-set comprehension method provides an improvement on our model. Our full model shows about 2% higher performance than the model without SSOC(TR+VAL). It is also interest-

Table 4.3: Results of ablation study about the occurrence flags. We demonstrate the accuracies of Text only, Diagram, and total questions without SSOC method.

Model	Text	Diagram	All
Our model w/o SSOC	<b>52.06</b>	<b>36.61</b>	<b>43.97</b>
w/o q-flag	49.29	35.78	42.21
w/o a-flag	43.24	31.50	37.09
w/o q & a-flag	42.64	31.72	36.92

ing to compare our full model with our model without SSOC(VAL). The results show that using the additional validation set on SSOC can improve overall accuracy compared to using only training set. It seems to have more advantage for learning unknown dataset in advance.

Our model without f-GCN & SSOC eliminates our two novel modules and replace GCN with vanilla RNN. That model shows 1% of performance degradation compared with the model without SSOC(TR+VAL) which means that it might not sufficient to deal with knowledge features with only RNN and attention module. Thus, context graph we create for each lesson could give proper representations with f-GCN module.

Table 4.3 shows the results of ablation study about occurrence flag. All models do not use SSOC method. In (4.5), we concatenate three components including the occurrence flag to create question or answer representation. We found that the occurrence flag which explicitly indicates the existence of a corresponding word in the contexts has a meaningful effect. Results of all types degrade significantly as ablating occurrence flags. Especially, eliminating a-flag drops accuracy about 7% which is almost 4 times higher than the decrease due to eliminating f-flag. We believe that disentangled features of answer candidates

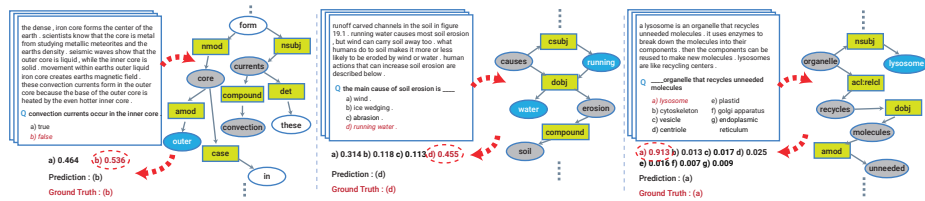


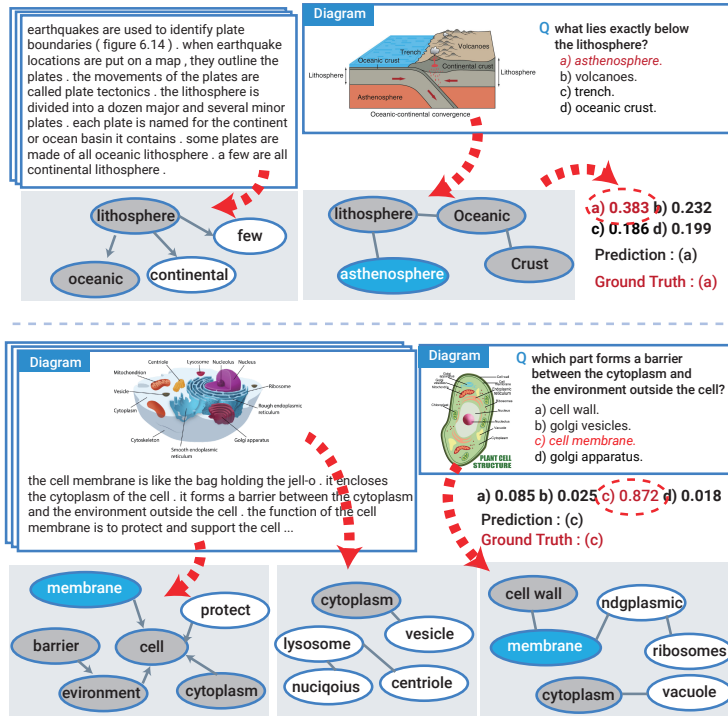
Figure 4.7: Qualitative results of text-type questions without visual context. Each example shows all items for a question in the textbook and a textual context subgraph to solve a question. And our predicted distribution for answers and ground truths are also displayed. In the subgraph, gray circles represent words in questions and blue circles represent words related to answers. Green rectangles represent relation types of the dependency graph.

can mainly determine the results while a question feature equally affects all features of candidates. Our model without both flags shows the lowest results due to the loss of representational power.

#### 4.4.5 Qualitative Results

Figure 4.7 shows three qualitative results of text-type questions without visual context. We illustrate textual contexts, questions, answer candidates and related subgraphs of context graphs.

The first example describes a pipeline on a T/F question. Three words, “currents”, “core” and “convection” are set as anchor nodes as shown in the left of Figure 4.7. Within two levels of depth, we can find “outer” node which is the opposite to “inner” in the question sentence. As a result, our model predicts the true and false probabilities of this question as 0.464 and 0.536, respectively, and correctly solves this problem as a false statement. Next example is a multi-



an important term, “lysosome” in choice (a). Therefore, choice (a) has a probability close to one among 7 candidates.

Figure 4.8 demonstrates qualitative results of diagram questions. We exclude relation type nodes in subgraphs of the dependency tree for simplicity and also illustrate diagram parsing graphs of visual contexts and question diagram. The example in the top shows intermediate results of subgraphs on a diagram question without visual context. Even though chosen paragraph in textual context do not include “asthenosphere”, graph of a question diagram contain relation between “asthenosphere” and “lithosphere”. Then our model can predict (a) as the correct answer with probability of 0.383. The bottom illustration describes the most complex case which has diagrams in both of context and question parts. We illustrate all subgraphs of text and diagrams. While our model can collect sufficient knowledge about cell structure on broad information scope, “cell membrane” can be chosen as correct answer with the highest probability.

In next pages, we present additional qualitative results of questions in three types. We explicitly demonstrates all intermediate results as subgraphs of visual context and question diagram. Note that we add a legend that indicates which types of data are used in this figure to avoid confusion. In Figure 4.9 and Figure 4.10, we illustrate intermediate and final results on text-type question with visual context. Next, we demonstrate intermediate and final results on diagram-type question without visual context in Figure 4.11 and Figure 4.12. Finally, we present intermediate and final results of the most complicated type, diagram-type question with visual context in Figure 4.13 and Figure 4.14. We hope the logical connectivity for solving the problem and how our model works well on the TQA problem are sufficiently understood with those figures.



These examples demonstrate abstraction ability and relationship expressiveness which can be huge advantages of graphs. Moreover, those results could support that our model can explicitly interpret the process of solving multi-modal QA.

## **4.5 Conclusion**

In this chapter, we proposed two novel methods to solve a realistic task, TQA dataset. We extract knowledge features with the proposed f-GCN and conduct self-supervised learning to overcome the out-of-domain issue. Our method also demonstrates state-of-the-art results. We believe that our work can be a meaningful step in realistic multi-modal QA and solving the out-of-domain issue.

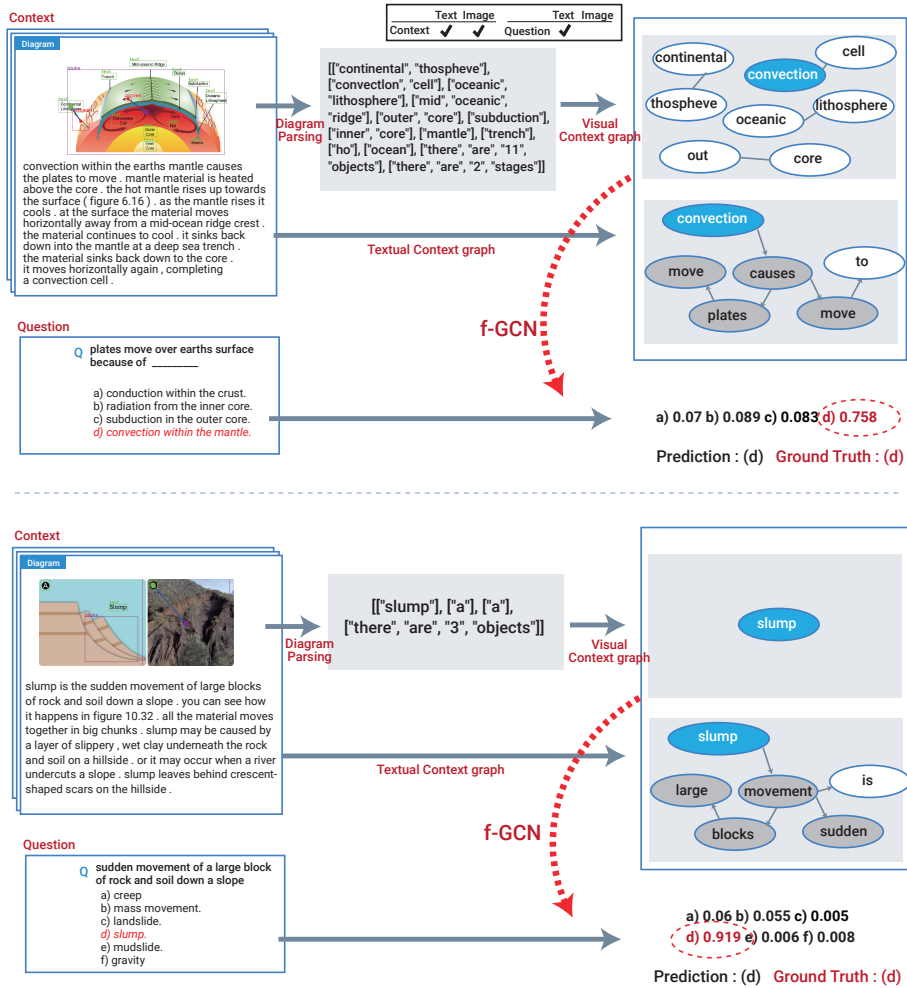


Figure 4.9: Additional qualitative results on text-type question with visual context. For both examples, a pipeline from visual context to visual context graph is shown. Gray circles represent words in questions and blue circles represent words related to answers.

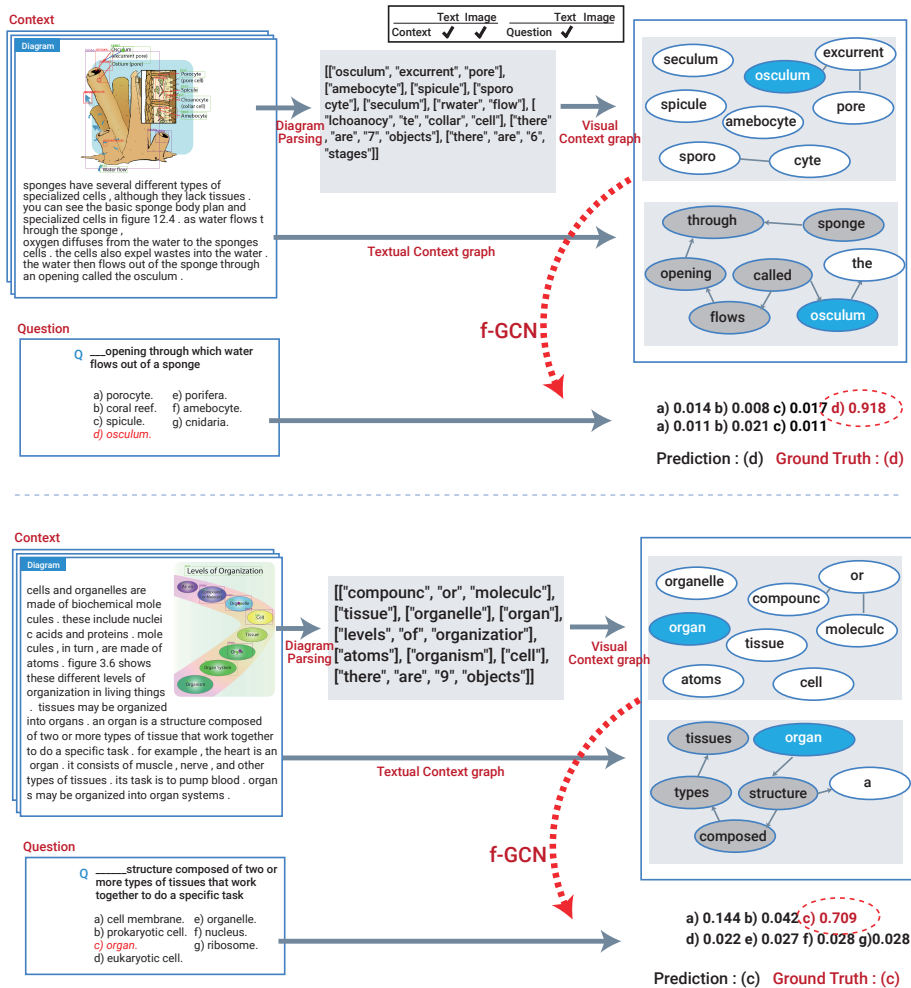


Figure 4.10: Additional qualitative results on text-type question with visual context. For both examples, a pipeline from visual context to visual context graph is shown. Gray circles represent words in questions and blue circles represent words related to answers.



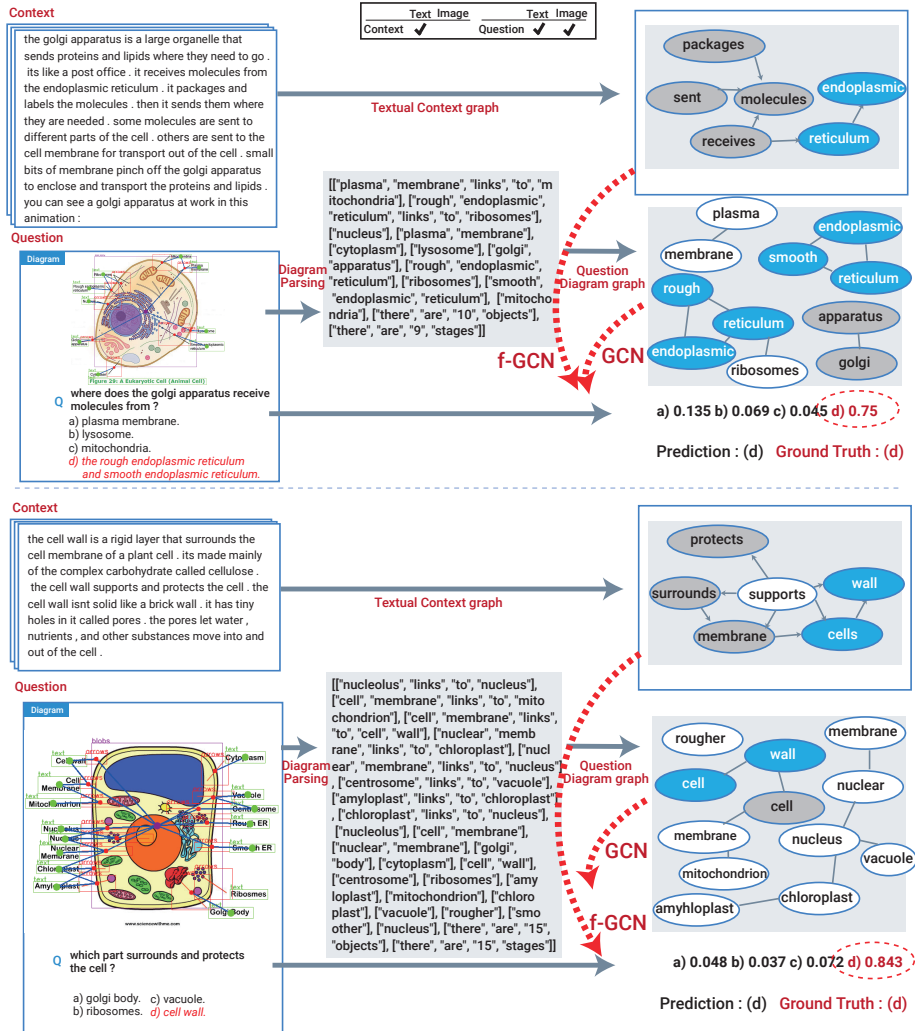


Figure 4.12: Additional qualitative results on diagram-type question without visual context. For both examples, a pipeline from question diagram to question

diagram graph is shown. Gray circles represent words in questions and blue circles represent words related to answers.

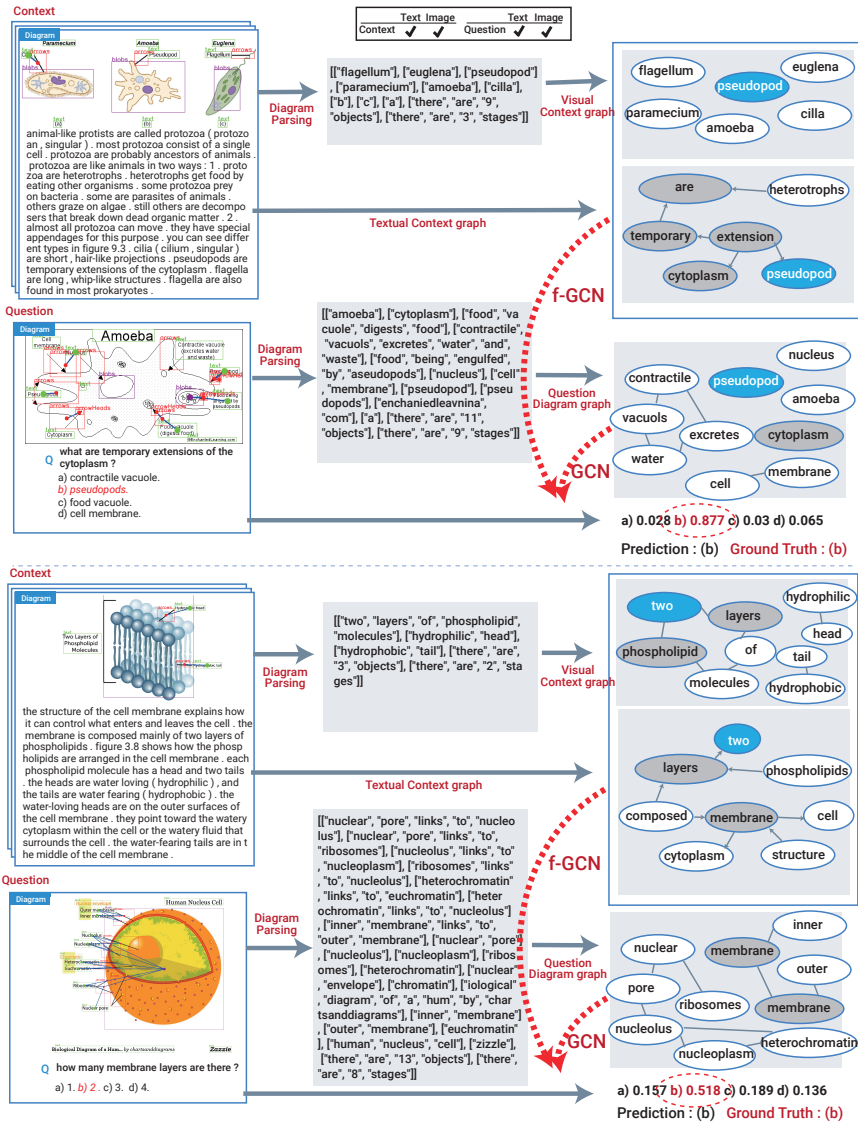


Figure 4.13: Additional qualitative results on diagram-type question with visual context. For both examples, pipelines from visual context and question diagram to visual context graph and question diagram graph are shown. Gray circles represent words in questions and blue circles represent words related to answers.

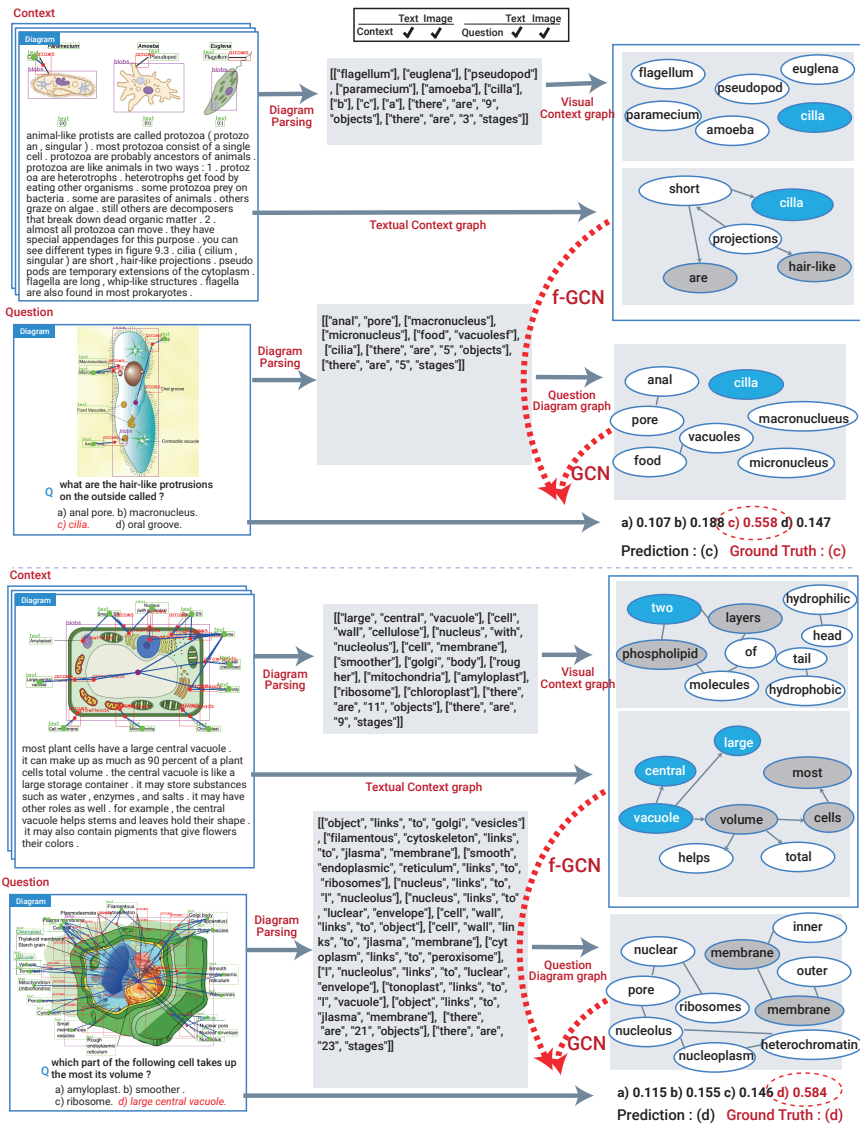


Figure 4.14: Additional qualitative results on diagram-type question with visual context. For both examples, pipelines from visual context and question diagram to visual context graph and question diagram graph are shown. Gray circles represent words in questions and blue circles represent words related to answers.

## Chapter 5

### **Weakly Supervised Object Detection with Human-object Interaction**

In prior chapters, while we can demonstrated that our methods with relational contexts work on diagrams, more interesting problems in real-life can be discussed in natural images. In particular, the relational context can help popular algorithms suffering fundamental issues. Among several successful algorithms, object detection has become one of the most successful fields in computer vision with various applications [57, 12, 56, 41]. Most of the successful models have emerged after the release of large scale datasets (e.g. PASCAL VOC, MS-COCO [15, 40]) with bounding box annotations. Given input images, conventional object detection models can localize boxes of objects and provide scores of object classes. Thus, they normally require manually annotated bounding boxes which have accurate coordinate values and object labels for training.

However, annotating bounding boxes is time-consuming and labor-intensive. It can also be difficult to expand the volume of a dataset by adding more object classes or adding more images. Therefore, researches to reduce those costs in



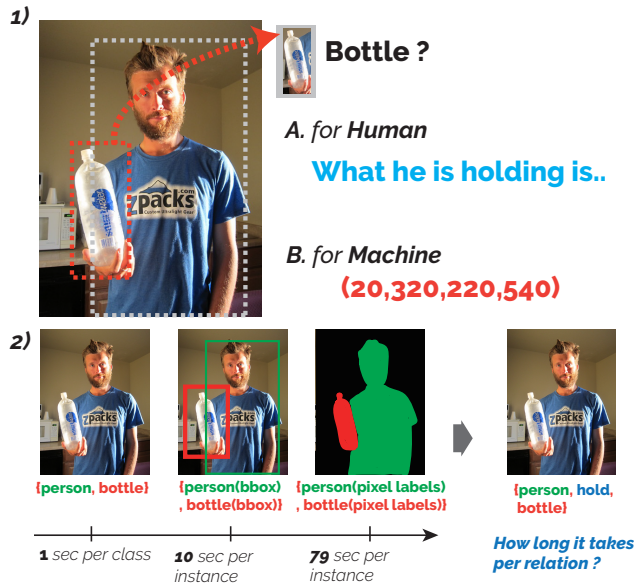


Figure 5.1: 1) Two different types of description of an object. A is human’s way of identifying an object while B is for machines. 2) Manually annotating time for three tasks. Bearman et al. [3] estimated annotation times for image level, bounding box and pixel level. At rightmost, annotating time of relation sentence can be similar to that of image-level since only action verb “hold” is added.

various ways have drawn attentions these days.

Weakly supervised object detection (WSOD) has been proposed to tackle aforementioned problems [81, 61, 25]. WSOD is to detect objects within images by weak supervision such as image-level labels. Despite the low annotation costs, the performance of WSOD is lower than that of full supervision.

To overcome a limitation of weak supervision, some approaches [61] rely on another type of full supervision with transfer learning. Transferred knowledge from a source domain could support weak supervision in a target domain. However, annotating other types of full supervision such as segmentation mask

labels is also expensive.

Our main intuition is that supervision of machines is totally different from that of humans. For example, Fig. 5.1 (1) shows different ways of identifying objects between humans and machines. While we should provide accurate coordinate values of object boxes for machines, a human usually recognizes new objects from contexts. Contexts also can reinforce supervision without much additional efforts.

Especially, how objects are related to human actions can be practical and advantageous since information about a human can be a proper evidence for recognizing contexts in an image. Moreover, humans can easily express contexts with sentences as shown in Fig. 5.1 (1), so that linguistic labels can be a key to reduce annotation cost for humans shown in Fig. 5.1 (2). Comparing to other annotating costs [3], cost of annotating a relation sentence such as “person”, “hold”, “bottle” can be almost similar to that of image-level annotation. Therefore, we propose a novel paradigm to learn unseen objects based on human-object interaction (HOI).

Our key idea is to exploit transferable knowledge from HOI contexts annotated as language as is in [8]. Hence we propose a novel module that predicts object locations from HOI. Since the actual coordinate values can not be specified, we use an attention map as localization results to connect it with a bounding box. Moreover, in order to train full object detector (e.g. Faster-RCNN [57]) in an end-to-end fashion, we design a new module as an add-on type.

The objective of this chapter is to make our model learn additional rare classes with weak verbal supervisions annotated easily by human. During the first stage, strong supervisions on non-rare classes teach our model to localize a proper location with a human pose and an action verb. In the next stage, only

weak supervisions with transferred knowledge keep training object detector for unseen rare classes. More details are described in Section 5.3.

The rest of this chapter is organized as follows: Prior works related to this problem are reviewed in Section 5.1. The overview of algorithm is illustrated in Section 5.2, and we propose detailed method in Section 5.3. Then, experimental results is demonstrated and discussed in Section 5.4. We conclude this chapter in Section 5.5.

## 5.1 Related Work

Most of the weakly supervised object localization and detection methods have been proposed based on an image-level supervision. With cheaper but weaker annotations, studies [4, 14, 28, 51, 63, 25] have mainly tried to enhance performance by multiple instance learning (MIL). In MIL, a bag is defined as a collection of regions in an image. It is labeled as positive if at least one object is positive and labeled as negative if all of the objects are negative. In the aspect of structure, Tang *et al.* [64] proposed the weakly supervised region proposal network to generate box proposals. On the other hand, methods in [10, 81] localize regions by generating object score heatmaps and determine bounding boxes around high score regions. To overcome the limitations of weak supervision, some research adapts transfer learning from another strong supervision. Shi *et al.* [61] have introduced things-and-stuff transfer, which learns a semantic segmentation on a source domain. The knowledge learned from things-and-stuff annotation is transferred to help localize objects in a target domain. While our work also exploits transferable knowledge, we use a cheaper supervision than the semantic segmentation. Since the costs of annotating relation classes of HOI

can be low, similar to image-level annotation (see Fig. 5.1), our model can be easily expanded to detect objects belonging to additional rare classes.

Yang *et al.* [74] have proposed activity-driven WSOD, which also exploits action classes as contextual information to localize objects without box annotations. However, this work [74] has generated box proposals using Selective Search [67], which is a conventional, rule-based algorithm. Since the box proposal method cannot be trained, there is a fundamental limitation which proper proposals hardly exist in novel data. Uijlings *et al.* [66] have addressed a new WSOD framework that revisits knowledge transfer for training object detectors on target classes. Since this work has optimized a box proposal network for target classes by MIL, box generators can be sufficiently trained with rare classes due to a lack of contextual information. Our method resolves the aforementioned issues with a novel box proposal module that can transfer knowledge using an HOI dataset.

## 5.2 Algorithm Overview

An overview of our algorithm is illustrated in Fig. 5.2. Let  $D = \{(I_i, y_i)\}_{i=1}^N$  is the data set, where  $y_i$  is the label of the image  $I_i$  and  $N$  is the number of images. The image label  $y_i$  is organized as a tuple as shown below:

$$y_i = \{(H_{verb}^j, O_{bbox}^j, O_{cls}^{j=1})\}_j^M \quad (5.1)$$

where,  $O_{bbox}^j, O_{cls}^j$  are the bounding box and the class of an object in the image,  $H_{verb}^j$  is the action verb corresponding to the object, and  $M$  is the number of tuples in the image  $I_i$ . To evaluate the proposed method, we divided  $D$  into two sub-categories based on the number of objects in a class: non-rare (source

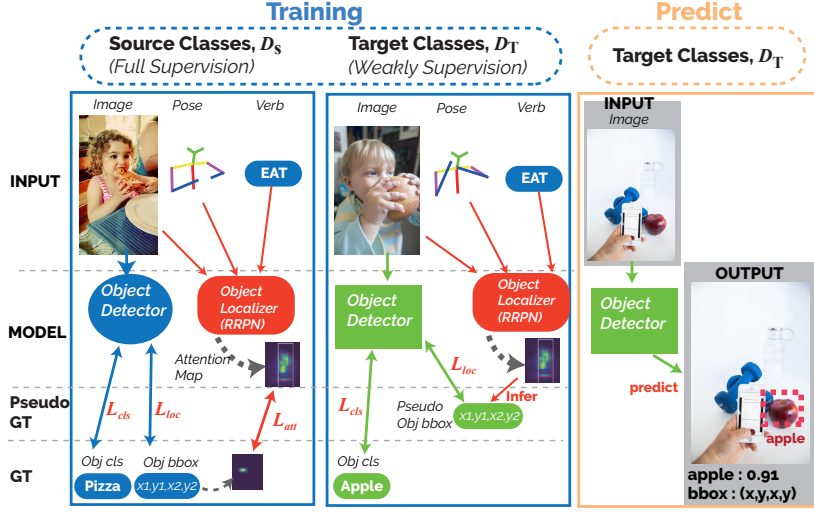


Figure 5.2: **Overview of our algorithm.** 1) During the training phase for source classes, RRPN is also trained to predict attention map from human-object interaction. 2) In the target-class training phase, an object detector is trained using the ground truth class label, and the box label provided by the trained RRPN. In other words, our problem focuses solely on solving the weakly supervised object detection problem on the target classes. 3) As a result, the trained object detector for target classes can infer box coordinates and object classes with only an image input.

classes)  $D_S$  and rare (target classes)  $D_T$ . Note that, there is no object class duplicates but all action verbs are overlapped between two subcategory datasets.

For source classes  $D_S$ , we normally train the first object detector (blue circle in Fig. 5.2) with full supervision using  $(O_{bbox}^j, O_{cls}^j)$ . Along with training of the object detector, we also train an object localizer (red circle) called RRPN with newly defined inputs. Since the RRPN should learn how to localize an object only with the information on a human and an action verb, we use the image

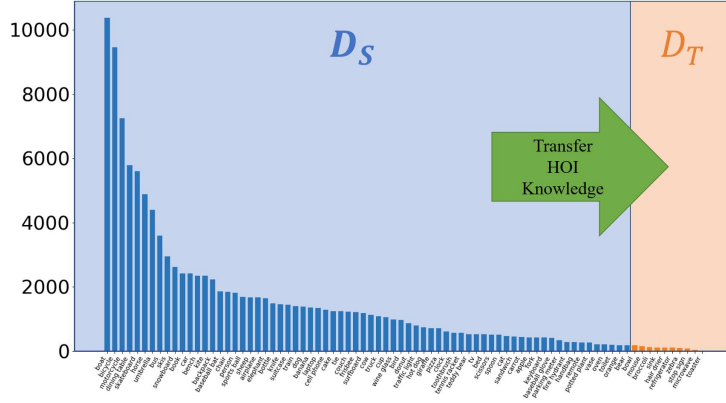


Figure 5.3: Categorization of source classes ( $D_S$ ) and target classes ( $D_T$ ) according to the number of objects in the training set of HICO-DET.

$I_i$ , the action verb  $H_{verb}^j$ , and the pose of the human  $H_{pose}^j$  as three inputs. The  $H_{verb}^j$  simply comes from  $y_i$  but the human pose  $H_{pose}^j$  is extracted from an image  $I_i$  with an existing human pose estimation method. As a results, the RRPN predicts an attention map  $\tilde{A}_i^j$  of an object location in the  $i$ -th image from human’s action and appearance. We optimize losses regarding the object class and the location using  $O_{bbox}^j$  and  $O_{cls}^j$  for the object detector, but create a Gaussian map of  $O_{bbox}^j$  and use it as a ground truth in the training of the attention map of the RRPN. In this phase, since object classes in  $D_S$  are non-rare, the RRPN can learn common knowledge between objects and human actions.

For target classes in  $D_T$ , only object class information  $O_{cls}^j$  and the action verb  $H_{verb}^j$  are available but there is no bounding box information. To fill the absence of  $O_{bbox}^j$ , we exploit learned knowledge inferred by the RRPN with the same kinds of inputs as the training phase for the source classes. Since the output of RRPN is an attention map, we extract a coordinate by thresholding it and generate pseudo bounding box  $\hat{O}_{bbox}^j$ . Then, we normally train the second

object detector (green rectangle in Fig. 5.2) for  $D_T$ . Since we already have used all action verbs to train the RRPN in the previous phase and transfer the same parameters in the training phase for the target classes, it can infer an object location with a human pose and an action verb. In Fig. 5.2, after the RRPN already learned to localize unseen object “*Apple*” with verb “*EAT*” and grabbing pose in the training phase of  $D_S$ , it can infer proper location as a pseudo ground truth  $\hat{O}_{bbox}^j$ . In conclusion, we use weak supervision by human actions to train a full object detector.

Eventually, the trained object detector in the second phase can predict objects in  $D_T$  only with an input image  $I_i$  as shown in Fig. 5.2. Although we have not shown the real location of “*Apple*”, it is possible to predict the class score and the coordinate of an “*Apple*” object.

In this scheme, we can additionally train new object detector for unseen rare classes without bounding box annotations. Moreover, since we already trained the RRPN with strong supervisions, we need smaller amount of data in target classes compared to other WSOD algorithms. Our experiments validate that we only use extremely rare object classes as target domain as shown in Fig. 5.3 to be trained sufficiently.

### 5.3 Proposed Method

Fig. 5.4 depicts the overall architecture of the proposed algorithm. The proposed algorithm consists of two modules, including the RRPN and the object detector. More precisely, it means that RRPN can be combined with the conventional architecture such as the Faster-RCNN. RRPN is a multi-stage encoder-decoder network, which is responsible for predicting an object-location-centric attention

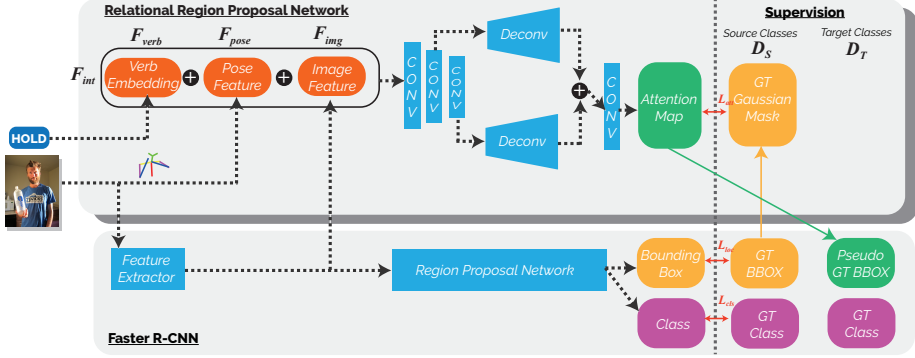


Figure 5.4: Overall network architecture of the proposed algorithm. Relational Region Proposal Network (RRPN) at the top is mounted on a basic Faster-RCNN model at the bottom. In RRPN, a combined feature  $F_{int}$  from verb, pose and image produces an attention map through a network which has four blocks. With source classes, the RRPN is trained with a Gaussian mask from the ground truth bounding box. However, with target classes, the RRPN generates a pseudo ground truth bounding box so that Faster-RCNN can be optimized.

map  $A_i^j$  from a multi-domain integrated feature map. The other module, object detector, is a conventional object detector which is trained for a given input image using the ground truth label  $O_{cls}$  and the bounding box  $O_{bbox}$ .

In order to exploit the knowledge of interaction, we train  $D_T$  after the training of  $D_S$  is done. Since, however, we do not account for the continual learning,  $D_S$  and  $D_T$  do not share parameters for the object detector. While the object detector is trained for  $D_S$  with supervision, at the same time, RRPN is also trained to learn the knowledge from interactions between a human and an object through action verbs (see Section 5.3.1). Then, the object detector is trained for the target classes without object bounding boxes, i.e. in a weakly supervised way, using the transferred knowledge from  $D_S$  (see Section 5.3.2).



### 5.3.1 Training on the Source classes $D_S$

Source classes are object classes on which data can be easily acquired. Training on the source classes is a standard supervised object detection procedure by using ground truth class and box labels for all the objects. The main purpose of training on the source classes is to predict an object location from a human-object interaction. Therefore, RRPN is also trained at the same time as the training of the object detector. The detailed training procedure for Faster-RCNN is applied in the same way as the original paper. The training procedure of RRPN is as follows.

#### Relational Region Proposal Network (RRPN)

RRPN is designed to be universally applicable to various task’s models, including other object detectors, in an add-on manner, and can share the backbone network with other model for image features to improve memory efficiency.

As mentioned above, RRPN predicts attention map  $\tilde{A}_i^j$  for a given image  $I_i$  using a multi-domain feature map  $F_{int}^{i,j} \in \mathbb{R}^{C \times H \times W}$  as an input, where  $C, W, H$  are the depth, width and height of the feature map. Multi-domain feature map is obtained by

$$F_{int}^i = \{F_{int}^{i,j}\}_{j=1}^M = \{(F_{img}^j \oplus F_{pose}^j \oplus F_{verb}^j)\}_{j=1}^M, \quad (5.2)$$

where,  $F_{img}$ ,  $F_{pose}$ ,  $F_{word}$  are the image feature, pose feature, and verb feature obtained by their corresponding models  $f_{img}(I_i, \theta_{img})$ ,  $f_{pose}(I_i, \theta_{pose})$ ,  $f_{word}(H_{act}^j, \theta_{verb})$ , and  $\oplus$  is the matrix concatenation. Here,  $\theta_x$  is the corresponding model parameters. The convolution operation for  $F_{int}^i$  computes the object existence probability for a combination of  $F_{img}^j$ ,  $F_{pose}^j$ , and  $F_{verb}^j$  at a specific location on the image  $I_i$ .

As in (5.2), we used three feature maps to utilize contexts from various domains in a given dataset, and each feature map has its own contribution. The pose and word feature are responsible for the visual context of the human’s location and action, and the distinguishable linguistic context for the human’s action, respectively. The image feature is responsible for representing the whole scene as well as the object of interest. The details of the model used for extracting the feature map in each domain are as follows:

**Pose feature** We use the well-known human pose estimation model, OPENPOSE [7], to extract pose features. OPENPOSE predicts the location of human body joints using image or video as an input. The output consists of channels corresponding to each joint and a channel representing background information. In this chapter, we used a pose estimation model with 19 channels including 18 joints and 1 background. In order to feed distinct information of human pose to the RRPN, we exploit the 18 channels except for the background channel as the pose feature.

**Verb feature** The widely used GloVe-twitter-27B-25d model [24] is applied as the word embedding model for the verb. Since a word is embedded into a vector, one needs to convert it into a tensor form for integration with other features. While  $F_{img}^j$  and  $F_{pose}^j$  may have different spatial-wise activations depending on  $I_i$ ,  $F_{verb}^j$  must have the same value regardless of positions. In designing  $F_{verb}^j$ , we also take this consideration into account. In order to match the spatial dimension with others, the verb feature is copied to every spatial position. So that dimension of  $F_{verb}^j$  is converted from  $\mathbb{R}^{25}$  to  $\mathbb{R}^{25 \times H \times W}$ . By stacking a depth-wise word vector at all spatial positions, we can conduct a convolution operation using the same verbal information at all position of  $F_{img}^j$ . Note that, among the HICO-DET datasets, tuples with ‘No interaction’ verb labels were excluded

from training and validation phases for accurate evaluation of the proposed algorithm.

**Image feature** The proposed algorithm makes use of the representative two-stage object detection model, Faster-RCNN. It consists of a feature extractor, a back-bone network, and a region proposal network (RPN). The output feature map of the back-bone network of the Faster-RCNN is used as the image feature for the RRPN. When training on the target classes, the parameters of the backbone network are reused, but the parameters of RPN are reset.

Multi-domain feature map  $F_{int}^{i,j}$  is then fed into the network to predict attention map  $A_i^j$ . In order to robustly detect objects in various sizes, we designed the network architecture which has four blocks as in [69]: an Encoder block  $f_{en}$ , two decoder blocks  $f_{de1}$ ,  $f_{de2}$  and an attention block  $f_{att}$ .  $f_{en}$  takes  $F_{int}^{i,j}$  as an input and outputs two feature maps with different spatial dimensions. Then, each output feature map feeds into  $f_{de1}$  and  $f_{de2}$ , respectively. The output feature maps of  $f_{de1}$  and  $f_{de2}$  having the same spatial dimension are concatenated and inputted to the attention block resulting in an attention map  $\tilde{A}_i^j$  as

$$\tilde{A}_i^j = f_{att}[f_{de1}\{f_{en}^1(F_{int}^{i,j})\} \oplus f_{de2}\{f_{en}^2(F_{int}^{i,j})\}]. \quad (5.3)$$

The output of RRPN is an attention map which emphasizes the location where the object likely to be located. To train attention maps, we create a Gaussian map  $A_i^j$ , as a ground truth attention map, using  $O_{bbox}^j$ . RRPN is trained using the ground truth attention map as the label. We use pixel-wise binary cross entropy loss (BCE) between two attention maps in (5.4)

$$L_{att} = BCE(A_i^j, \tilde{A}_i^j). \quad (5.4)$$

The total loss for training on the source classes including RRPN and object

detector is shown below:

$$L_{total} = L_{det} + \lambda L_{att}, \quad L_{det} = L_{cls} + L_{loc} \quad (5.5)$$

where,  $\lambda$  is a hyper-parameter balancing between the two losses and  $L_{det}$  is the loss for the Faster-RCNN. In the object detector point of view, the proposed algorithm on the source classes is trained in the same way as the conventional supervised object detection algorithms.

### 5.3.2 Training On the Target classes $D_T$

The object detector for the target classes should be trained without  $O_{bbox}$ . Therefore, we define this problem as a weakly supervised object detection (WSOD) problem. We use  $\tilde{O}_{bbox}$  as an alternative to the missing  $O_{bbox}$  utilizing RRPN learned in the source classes training phase. It is expected that the trained RRPN can predict locations of unseen objects i.e. target classes, since it is trained to predict the object location using a human pose, an action (verb) and an image feature. The training process on the target classes using the trained RRPN is as follows:

The human pose, verb, and image features are fed into the trained RRPN. We apply a threshold to obtain a pseudo bounding box from the output attention map as

$$\tilde{A}_i^j = \begin{cases} 1, & \text{if, } \tilde{A}_i^j > \delta \\ 0, & \text{otherwise} \end{cases} \quad (5.6)$$

where,  $\delta$  is a pre-defined threshold. The largest bounding box containing a valid value in  $\tilde{A}_i^j$  is called  $\tilde{O}_{bbox}^j$ . The pseudo ground truth bounding boxes  $\{\tilde{O}_{bbox}^j\}_{j=1}^M$  obtained from the attention maps  $\{\tilde{A}_i^j\}_{j=1}^M$  of all tuples in the image  $I_i$  are collected together and used as bounding box labels  $\tilde{O}_{bbox}$  for training

an object detector. In this step, a different type of object detector from the one trained in the source-class training phase can be used for training. The object detector trains to minimize detection loss using  $O_{cls}$  and  $\tilde{O}_{bbox}$ .

## 5.4 Experiment

In this section, we evaluate the performance of the proposed WSOD algorithm. To the best of our knowledge, no previous studies have been conducted on the relationship between object detection and HOI. Thus, we omit the performance comparison with a baseline and focused on analyzing the proposed algorithm.

### 5.4.1 Implementation details

The overall structure of the proposed WSOD method consists of RRPN and Faster-RCNN. We have used Faster-RCNN with ImageNet pre-trained ResNet-101 model. RRPN consists of  $f_{img}(I_i, \theta_{img})$ ,  $f_{pose}(I_i, \theta_{pose})$ , and  $f_{word}(H_{act}^j, \theta_{word})$ , and each model consists of backbone of Faster-RCNN, OPENPOSE<sup>1</sup> [7], and GloVe<sup>2</sup> [24] as described previously. We have used the pre-trained OPENPOSE and GloVe models only to extract each feature without further training.

The spatial dimension of the integrated feature map is equal to the output feature map of the Faster-RCNN backbone network, i.e.  $40 \times 40$ .  $f_{en}$ ,  $f_{de1}$  and  $f_{de2}$  consist of 3, 5, 6 convolutional layer, respectively. Max pooling and appropriate strides were used to fit the corresponding spatial dimension.  $f_{att}$  is  $1 \times 1$  convolution layer followed by a sigmoid layer. The spatial dimension of the output feature map is the same as the input spatial dimension of the object

---

<sup>1</sup><https://github.com/CMU-Perceptual-Computing-Lab/openpose>

<sup>2</sup><https://github.com/stanfordnlp/GloVe>

detector. Hyper parameters for training RRPN is as follows: learning rate:  $1e-3$ , optimizer: stochastic gradient descent, weight decay:  $1e-4$ , momentum: 0.9, batch size: 4, epoch or iteration: 15 epoch (source class), 30 epoch (target class). Hyper parameters for Faster-RCNN is set as the same as the original paper.

### 5.4.2 Dataset and Pre-processing

HICO-DET dataset consists of 47,776 images (38,118 training and 9,658 testing) classified into 117 actions (verb) and 80 object classes, and the object classes are the same as MS-COCO dataset. The ground truth labels consist of a tuple of  $(H_{act}, O_{bbox}, O_{cls})$  as in (5.1). Note that, the RRPN is trained based on tuples, so images containing multiple tuples are fed multiple times. The total number of tuples is 151,276 (117,871 training and 33,405 testing), and we use 131,560 tuples (102,450 training and 29,110 testing) excluding the tuples corresponding to the action label ‘no-interaction’.

In order to construct the problem environment in this chapter, the whole dataset is divided into source and target datasets according to the frequency of the object class as shown in Fig. 5.3. Our basic experiment is set up with 116 verbs excluding ‘no interaction’, 70 source classes and 10 target classes.

We also verified the proposed algorithm on V-COCO dataset for qualitative analysis. The purpose of evaluation on the V-COCO dataset is to show that the knowledge can be transferred from one dataset to other. Details on both datasets are described in the supplementary material.

### 5.4.3 Metrics

We use mean Average Precision (mAP) and Recall as evaluation metrics. Because RRPN produces one bounding box for one tuple (action), Recall is used

Table 5.1: Comparison of quantitative result of different feature combination. (I : Image feature, P : Pose feature, V: Verb feature,  $D_S$  : Source,  $D_T$  : Target, W : Weakly supervised object detection, S : Supervised object detection,  $\lambda = 10$ ,  $\delta = 0.1$ )

I	P	V	Recall@.5 (RRPN)		mAP@.5 (Faster-RCNN)		
			$D_S$ (%)	$D_T$ (%)	$D_S$	$D_T(W)$	$D_T(S)$
✓	✓	✓	<b>47.69</b>	<b>28.64</b>	<b>30.34</b>	<b>17.19</b>	29.37
✓			42.00	22.75	23.57	9.57	22.07
✓	✓		41.42	22.13	24.17	10.07	25.15
✓		✓	46.34	23.84	29.97	16.34	<b>30.28</b>

to measure how accurate the location of an object corresponding to an action is. In other words, Recall evaluates the objectness of  $\tilde{A}$  predicted by RRPN, and is calculated as the ratio of tuples for which  $\text{IoU} > 0.5$ . On the other hand, the object detector detects all the objects in an image at once. Therefore, we use the mAP in measuring the performance of Faster-RCNN which are the standard metrics for object detectors.

The source and target in Recall are the performance of the RRPN’s agent after the source training. When training the target classes, RRPN is fixed and not trained. Note that Recall is measured on test set for  $D_S$  and on both training and test set for  $D_T$ .

#### 5.4.4 Comparison with different feature combination

We experiment to verify the performance of different feature combinations. We train and test the RRPN using the same types of feature for both the source and

Table 5.2: Comparison of quantitative result of different attention loss balance and box threshold (  $D_S$  : Source,  $D_T$  : Target, W : Weakly supervised object detection, S : Supervised object detection)

parameter		Recall@.5 (RRPN)		mAP@.5 (Faster-RCNN)		
$\lambda$	$\delta$	$D_S$ (%)	$D_T$ (%)	$D_S$	$D_T(W)$	$D_T(S)$
0	0.1	11.64	11.19	<b>31.06</b>	1.61	25.57
1	0.1	41.51	24.46	23.87	9.27	25.43
5	0.1	46.37	23.37	30.10	14.38	26.11
10	0.1	<b>47.69</b>	<b>28.64</b>	30.34	<b>17.19</b>	<b>29.37</b>
15	0.1	46.65	23.07	23.32	15.85	25.45
20	0.1	43.17	26.00	22.68	15.75	22.92
10	0.05	<b>48.22</b>	<b>29.41</b>	30.27	9.41	25.04
10	0.10	47.69	28.64	30.34	<b>17.19</b>	29.37
10	0.15	39.67	17.96	<b>30.40</b>	14.01	26.66
10	0.20	34.14	16.72	30.22	13.24	<b>30.39</b>

the target in each experiment. Table 5.1 shows the performances of RRPN and Faster-RCNN as Recall and mAP, respectively, using different combinations of features. The  $D_T(W)$  in mAP is the results of our WSOD, and the  $D_S$  and  $D_T(S)$  are the results of full supervision.

Our full model combining all three features in the top of Table 5.1 shows the highest performance in both Recall and mAP among all combinations. The mAP of  $D_T(W)$  has 17.19%, which is 7.62% better than image-only model and the Recall of the  $D_T$  is 28.64% which is about 5% higher than other combinations. Moreover, the mAP score of our full model is only 4.88% lower than image-



Table 5.3: Comparison of the mAP with other WSOD algorithms on HICO-DET. (PCL\* is tested by ourselves, § is trained on the entire dataset and † is trained on  $D_S$  and  $D_T$  separately. I : Image feature, P : Pose feature, V: Verb feature,  $D_S$  : Source,  $D_T$  : Target, W : Weakly supervised object detection, S : Supervised object detection,  $\lambda = 10$ ,  $\delta = 0.1$ )

Methods	AD [74]	PCL [62]	PCL*		Ours (I)	Ours (I+P+V)
$D_S(W)$	-	-	4.80 <sup>§</sup>	5.01 <sup>†</sup>	-	-
$D_T(W)$	-	-	0.01 <sup>§</sup>	4.75 <sup>†</sup>	9.57	17.19
Total	5.39	3.62	4.42 <sup>§</sup>	-	-	-

only fully supervised model in  $D_T(S)$ . Compared to models of full supervision  $D_T(S)$ , we believe that the mAP score of our full model can meaningfully show that it can be trained despite weak supervision of rare classes.

In the middle, using  $f_{img}$  alone, Recall in the target has 22.75% and mAP ( $D_T(W)$ ) are much lower than mAP ( $D_T(S)$ ). It means that RRPN could not be trained solely by  $f_{img}$ . The two results in the bottom are the performance with two features combined. When the  $F_{pose}$  is combined with the  $F_{img}$ , Recall degrades and the mAP increases slightly. It can show that  $F_{pose}$  that is extracted from an image is redundant unless it interacts with a verb. Combining  $F_{verb}$  with  $F_{img}$ , however, Recall and mAP significantly increase and the mAP of  $D_T(S)$  It is interesting that it might be more effective for not only RRPN but also Faster-RCNN when using combination of features from other domain.

### 5.4.5 Comparison with different attention loss balance and box threshold

In experiments in Table 5.2, we focus on verifying the effect of  $\lambda$  in (5.5) and  $\delta$  in (5.6) on the performance in terms of a shared parameter.

In top of Table 5.2, according to the change of the  $\lambda$  in (5.5), the ratio of the loss weight in RRPN is determined. When  $\lambda$  is zero, due to untrained RRPN, Recall and mAP for  $D_T(W)$  have the lowest score while mAP for  $D_S$  has the highest score. On the other hand, Recall and mAP are the highest at  $\lambda = 10$  with performance improvements of 17.45% and 15.58% compared to  $\lambda = 0$ , respectively. On the contrary, on some levels of  $\lambda$ , we can see that the performance degradation for not only  $D_T(W)$  but also  $D_S$ .

This can be understood as an effect of parameter sharing for image feature extractor between RRPN and an object detector. As mentioned earlier, RRPN is a universal add-on type module which can be adapted to various computer vision tasks. To effectively utilize these advantages, we share the backbone network of RRPN and the object detector in consideration of memory efficiency. Therefore, the RRPN and the object detector affect each other through the backbone network during training.

In bottom of Table 5.2, according to the  $\delta$  in (5.6), the size of the pseudo bounding box is determined. A small  $\delta$  makes the size of the boxes increase, while a large  $\delta$  makes the box small or disappear. As  $\delta$  increases, partial information of the object is trained. For example, in the case of an apple, only the central part of the apple is trained with high  $\delta$ , which causes many false positive. On the other hand, lowering the threshold of the box, Faster-RCNN is trained not only with an object but also with backgrounds. It is interesting that  $\delta$  affect

differently to both metrics where Recall gets higher when  $\delta$  gets smaller but mAP get the highest score when  $\delta = 0.1$ . We believe that the RRPN can easily learn objectness with a larger box due to small  $\delta$ , but classification of objects could be more difficult due to inaccurate localization. Therefore, too small or too large  $\delta$  causes a degradation of mAP, and we have found the suitable value,  $\delta = 0.1$ , through the experiments by selecting the value with the highest performance in  $D_T(W)$ .

#### 5.4.6 Comparison with prior works

Lastly, we conduct experiments to compare with prior works on HICO-DET as shown in Table 5.3. First two columns represent overall results of original algorithms in [74] and [62]. However, both results are only able to show performance of all object classes with an entire dataset. Since our method is designed for transfer learning, we experiment to validate PCL algorithm on each of source and target domains. As a result, our best model with image, pose and verb has 17.19% which is 4 times better than a result of PCL on  $D_T$ . Moreover, our model only with the image feature outperforms PCL algorithm on  $D_T$ . Although a direct apple-to-apple comparison is difficult, we can see that our method is far better than the compared methods.

#### 5.4.7 Qualitative results

Note that experiments for the qualitative results are conducted on a smaller dataset compared to the basic set up for the quantitative results. For qualitative result, we use 5 verbs and 10  $D_S$  and 70  $D_T$ , other hyper-parameters are the same as the basic set up. The reason why we shrink the size of the dataset for the qualitative result is for representing the effectiveness of RRPN more clearly.

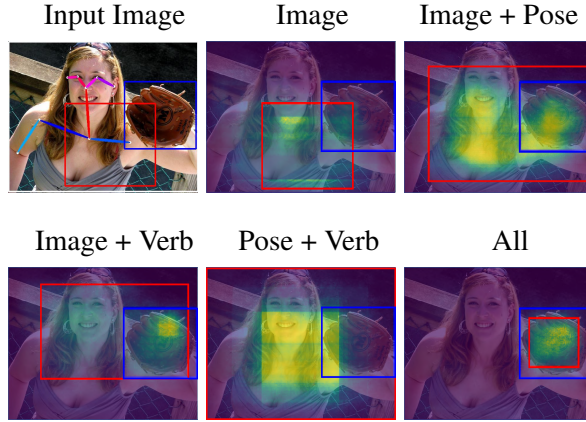


Figure 5.5: Comparison of predicted attention maps trained only by the image feature and by various integrated features with [glove, hold]. One can see that the predicted attention maps show different activations depending on the role of each feature map.

Fig. 5.7 and Fig. 5.8 show the qualitative results of the proposed algorithm on  $D_T$ . The first column indicates input images and the last column indicates output attention maps inferred by the corresponding actions. We can see that RRPN predicts an accurate attention map on unseen object classes in  $D_T$ . Furthermore, it can be seen that the pattern of the predicted attention map differs depending on the verb. For example, while ‘*hold*’ shows a strong activation value near the human hand, ‘*ride*’ tends to activate at the bottom of a person. Based on this, we can confirm that the object location can be estimated based on the interaction between the verb and the pose. The role of the pose can be found in the example of [Giraffe, Ride]. Despite that two giraffes exist in an image, the activation of a giraffe on which the human is riding shows slightly stronger than the other. This can be seen as a contribution of the human pose

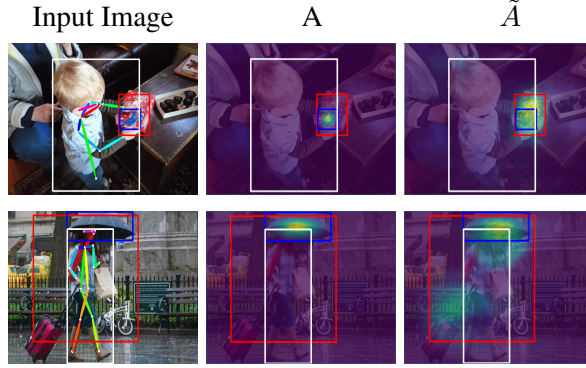


Figure 5.6: Unsuccessful results. Box color notation is the same as Fig. 5.7. Top and bottom rows are the results of [hold, scissors] and [hold, umbrella]. Inaccurate object localization occurs when (top) two objects are overlapping (bottom) or the human doing the same action for multiple objects.

feature to the object localization. We also verified the performance of RRPN on the object from a different dataset, V-COCO. The RRPN is trained using  $D_S$  of HICO-DET and predicts the attention map of  $D_T$  of V-COCO. The bottom row shows the predicted attention map on V-COCO. We can see that the proposed algorithm can also predict the object location accurately on images even from other datasets.

In addition, Fig. 5.9 and Fig. 5.10 show the qualitative results on  $D_T$  especially regarding *carry* and *sit on*. The pattern of the predicted attention maps differs depending on the verb: while *carry* focuses on near the human hand, *sit on* activates at the bottom of a human. These results are analogous to results of *hold* and *ride* in previous results.

Fig. 5.5 depicts the comparison of predicted attention map between different feature combinations on [glove, hold]. As described in section 5, the pattern

of the resulting attention map can be changed by the combination of features. Since “Glove” is an unseen class, backbone has no information to extract reliable feature, so that RRPN cannot predict the location of an object accurately using only  $F_{img}$ . However, if RRPN is trained using more than two features including  $F_{img}$ , RRPN can infer the location of an object based either on  $F_{pose}$  or on  $F_{verb}$ . Specifically,  $F_{img} + F_{verb}$  predicted a more distinguishable attention map for an object, compared to  $F_{img} + F_{pose}$  feature map. Since  $F_{img}$  and  $F_{pose}$  are extracted from the same image, some of the information can be redundant between two features. On the other hand,  $F_{verb}$  is able to provide useful information to  $F_{int}$  because it is extracted from a different domain, language. Consequently, the location of an object can be predicted precisely when we use all three features. On the contrary, if RRPN trained using only  $F_{pose}$  and  $F_{verb}$  without  $F_{img}$ , the output attention map only activates around the human. Thus, it can be understood that  $F_{verb}$  plays a role of providing supplementary information to  $F_{img}$  about the object of interest.

Fig. 5.6 shows some examples of unsuccessful results. The verb and object class labels are annotated as [hold, scissors] and [hold, umbrella], respectively. The predicted attention maps are focused on the box instead of the scissors, and on both umbrella and suitcase rather than only the umbrella. Since RRPN predicts the location of an object according to human and its corresponding action, inaccurate attention maps can be generated when multiple objects are involved in the same action. Qualitative results are further described in the supplementary material.

## 5.5 Conclusion

In this chapter, we proposed a novel weakly-supervised scheme for object detection problems. We introduced the RRPN which can universally localize objects in an image with information on human poses and action verbs. Using transferable knowledge from the RRPN, we can continuously train any object detector for unseen objects with weak verbal supervision describing HOI. We validated our method based on the results on HICO-DET dataset and the performances show the possibility of our method for a new WSOD training scheme. Our work shows sufficient potentials to overcome the inefficiency of the supervised training scheme in recent deep learning. Also, we can develop our method in the direction to the continual learning since we already suggested a novel method to transfer common knowledge to localize objects with HOI.

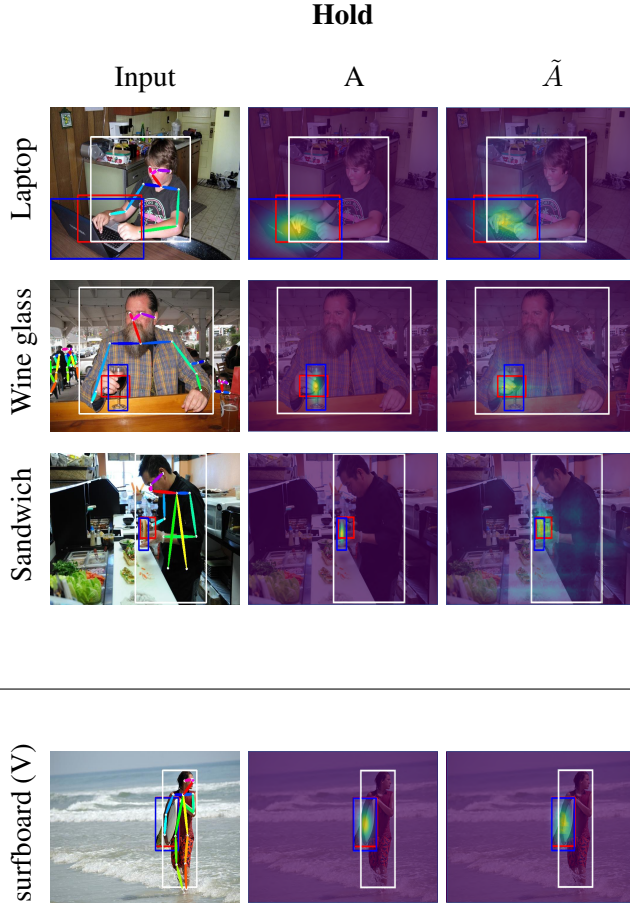


Figure 5.7: (Left) Input image with pose, (middle) ground truth Gaussian attention mask (A) in yellow, and (Right) predicted attention map ( $\tilde{A}$ ). Red box is pseudo object box, blue box is ground truth and white box indicates the human in action. (V) the last row is the result on the V-COCO dataset. Note that a white box is used solely for visually representing an acting human in an image and is not used in training on the target class  $D_T$ .



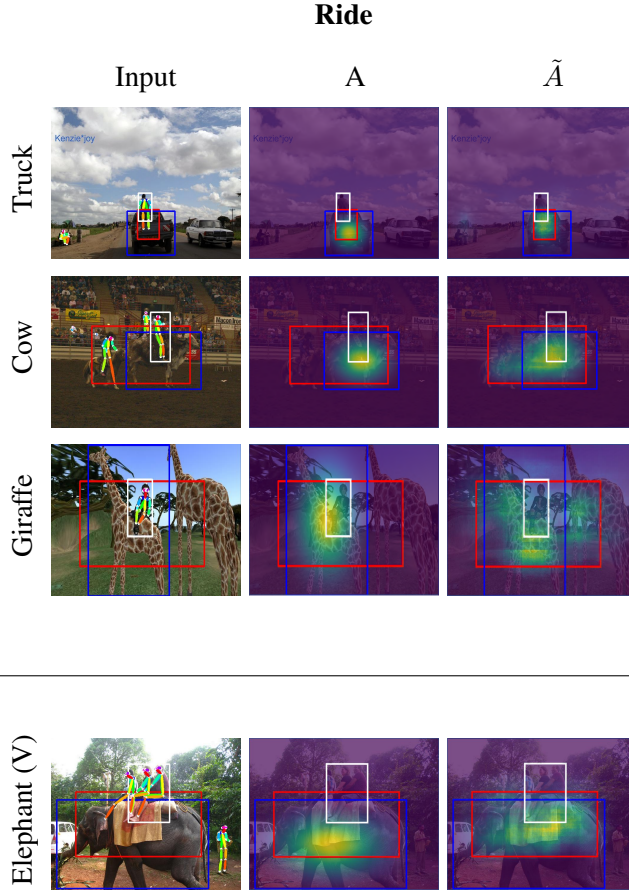


Figure 5.8: (Left) Input image with pose, (middle) ground truth Gaussian attention mask (A) in yellow, and (Right) predicted attention map ( $\tilde{A}$ ). **Red box** is pseudo object box , **blue box** is ground truth and white box indicates the human in action. (V) the last row is the result on the V-COCO dataset. Note that a white box is used solely for visually representing an acting human in an image and is not used in training on the target class  $D_T$ .

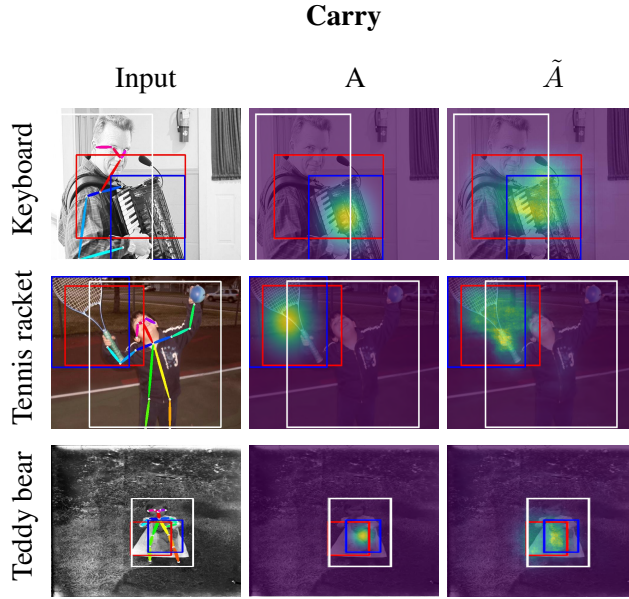


Figure 5.9: (Left) Input image with pose, (middle) ground truth Gaussian attention mask (A) in yellow, and (Right) predicted attention map ( $\tilde{A}$ ). **Red box** is pseudo object box , **blue box** is ground truth and white box indicates the human in action. Note that a white box is used solely for visually representing an acting human in an image and is not used in training on the target class  $D_T$ .

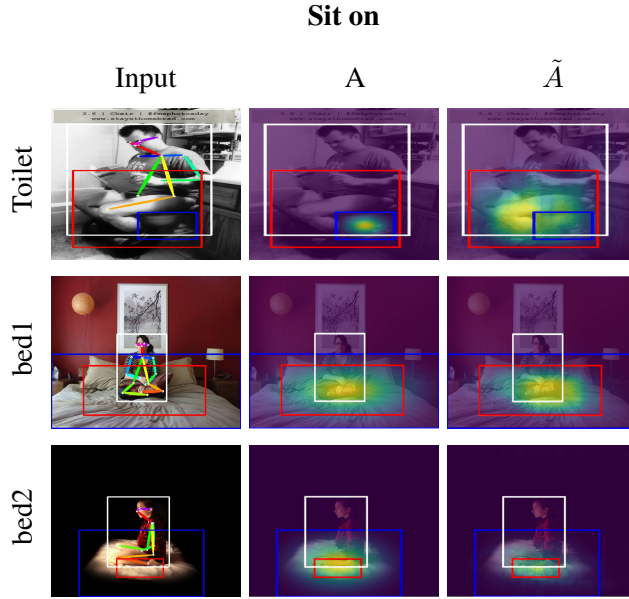


Figure 5.10: (Left) Input image with pose, (middle) ground truth Gaussian attention mask (A) in yellow, and (Right) predicted attention map ( $\tilde{A}$ ). **Red box** is pseudo object box , **blue box** is ground truth and white box indicates the human in action. Note that a white box is used solely for visually representing an acting human in an image and is not used in training on the target class  $D_T$ .

## **Chapter 6**

### **Concluding Remarks**

In this dissertation, we address three problems and propose novel methods for each task with semantic relations. In this chapter, we give a brief summary of the methods proposed in this dissertation. We then discuss limitations and future directions of our research.

#### **6.1 Summary**

Although recent and rapid advances have encouraged an expansion of research topics in the deep learning field, research is still focused on enhancing the performance of recognition algorithms. Among recent emerging topics, it is crucial to research the semantic image understanding that encourages machines to understand the meaning of a visual scene as a human does. When endeavoring to understand a visual scene semantically, relationships between objects can play an important role in providing contextual information. In this dissertation, we solved three tasks which involve visual relations.

First, we addressed the construction of semantic relation graphs in diagrams.

We proposed a novel end-to-end framework to predict graphs at once to avoid accumulated errors. We also suggested a new module called DGGN, which fully exploits the information from graph structures dynamically.

Second, we solved the most complicated QA problem with the multi-modal context graph. Since a TQA dataset includes both textual and visual contexts, we have proposed the f-GCN, which fuses a multi-modal graph into integrated features. Moreover, to overcome an out-of-domain issue, we pretrained the network in self-supervised way.

Third, we developed a weakly supervised object detection paradigm that can train object detectors for unseen classes without much effort. The word vector of the verb, which addresses a relation between a human and an object, can provide crucial information about understanding a scene. Therefore, our novel module called RRPN can learn and transfer contextual information to estimate the location of objects. In conclusion, we can continuously obtain a new object detector for additional, rare object classes

## **6.2 Limitations and Future Directions**

All of the methods proposed in this dissertation are focused on the exploitation of visual relations. Since our work dealt with challenging issues, some limitations exist; hence, future research directions can be discussed.

Inferring relationships in images is inherently a primitive research field that presents a number of challenges. For instance, in our first task, arbitrary layouts of diagrams led to the degradation of experimental results. Inferring relationships also needs to overcome the ambiguity of relations in natural images to exploit knowledge in various ways. Most of the research related to visual rela-

tions has tried to obtain relationships of pairs of objects to reduce the complexity of given problems. However, there are many situations in which more than two objects can be related to each other to represent contexts in images

A lack of datasets that represent the semantic relationship in images is also a limitation that weakens the generality of the proposed algorithms. Most of the datasets used in this dissertation have a relatively small number of images compared to datasets of other mature domains. A model learned from these datasets may work poorly and contain arbitrary relationships and contexts that did not appear in the datasets. Recently, a few datasets [78, 68] with dense annotation for semantic relationships in movies and real worlds have been released. Due to the difficulties of manual annotation for semantic labels, we expect more large datasets to be released to stimulate further research of semantic understanding.

There are several directions for future research to achieve better semantic image understanding. There are various approaches to teach a machine to understand visual inputs semantically like a human does. Therefore, a fundamental issue is how to define the problem to mimic humans' perception of semantic understanding. Below, we discuss some suggestions for defining problems.

One promising direction is to use the structural approach to mimic the logical process of human understanding. While a low-level of perception in the brain is hardly known as a structure, a higher level of perception, such as logic or language, can work based on structural knowledge. Therefore, more structural approaches should be beneficial in contributing to advances of semantic image understanding when defining research problems.

Moreover, another direction is to focus on the interpretability of methods for semantic understanding. Since recent studies have designed methods to use the implicit power of neural networks as a black box, methods has focused on how

to build an architecture of networks to solve specific problems. However, for a structural approach, we should design an explicit process to visualize logical evidence for high-level perceptions. Hence, we believe that solving problems using tools such as graphs, sentences and attention maps may be useful for solving semantic image understanding.

Lastly, in order to exploit contextual knowledge from semantic relations for further learning, we should resolve the continual learning issue. In our last task, we incorporated each object detector for each domain to train object classes sufficiently. For general artificial intelligence (GAI), our model should continuously learn new classes containing unseen data with transferred known knowledge. We believe that research for continual semantic understanding will have a huge impact on the deep learning domain.

# Bibliography

- [1] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Batra, and D. Parikh. Vqa: Visual question answering. *arXiv preprint arXiv:1505.00468*, 2015.
- [2] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multi-scale combinatorial grouping. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 328–335, 2014.
- [3] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. What’s the point: Semantic segmentation with point supervision. In *European conference on computer vision*, pages 549–565. Springer, 2016.
- [4] H. Bilen and A. Vedaldi. Weakly supervised deep detection networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2846–2854, 2016.
- [5] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. AcM, 2008.
- [6] J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun. Spectral networks and locally connected networks on graphs. In *International Conference on*



*Learning Representations (ICLR2014), CBLS, April 2014, 2014.*

- [7] Z. Cao, T. Simon, S. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *CVPR*, 2017.
- [8] Y.-W. Chao, Y. Liu, X. Liu, H. Zeng, and J. Deng. Learning to detect human-object interactions. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 381–389. IEEE, 2018.
- [9] Y.-W. Chao, Z. Wang, Y. He, J. Wang, and J. Deng. Hico: A benchmark for recognizing human-object interactions in images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1017–1025, 2015.
- [10] P. Dabkowski and Y. Gal. Real time image saliency for black box classifiers. In *Advances in Neural Information Processing Systems*, pages 6967–6976, 2017.
- [11] B. Dai, Y. Zhang, and D. Lin. Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3076–3086, 2017.
- [12] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016.
- [13] M. Defferrard, X. Bresson, and P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems*, pages 3844–3852, 2016.
- [14] A. Diba, V. Sharma, A. Pazandeh, H. Pirsiavash, and L. Van Gool. Weakly supervised cascaded convolutional networks. In *Proceedings of the IEEE*

- conference on computer vision and pattern recognition*, pages 914–922, 2017.
- [15] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.
  - [16] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*, 2016.
  - [17] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
  - [18] G. Gkioxari, R. Girshick, P. Dollár, and K. He. Detecting and recognizing human-object interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8359–8367, 2018.
  - [19] M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734. IEEE, 2005.
  - [20] A. Gupta and L. S. Davis. Objects in action: An approach for combining action understanding and object perception. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.
  - [21] A. Gupta, A. Kembhavi, and L. S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(10):1775–1789, 2009.

- [22] W. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, pages 1024–1034, 2017.
- [23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [24] R. S. J. Pennington and C. Manning. Glove: Global vectors for word representation. *conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
- [25] Z. Jie, Y. Wei, X. Jin, J. Feng, and W. Liu. Deep self-taught learning for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1377–1385, 2017.
- [26] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 1988–1997. IEEE, 2017.
- [27] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3668–3678, 2015.
- [28] V. Kantorov, M. Oquab, M. Cho, and I. Laptev. Contextlocnet: Context-aware deep network models for weakly supervised localization. In *European Conference on Computer Vision*, pages 350–365. Springer, 2016.

- [29] A. Kembhavi, M. Salvato, E. Kolve, M. Seo, H. Hajishirzi, and A. Farhadi. A diagram is worth a dozen images. In *European Conference on Computer Vision*, pages 235–251. Springer, 2016.
- [30] A. Kembhavi, M. Seo, D. Schwenk, J. Choi, A. Farhadi, and H. Hajishirzi. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5376–5384. IEEE, 2017.
- [31] D. Kim, Y. Yoo, J.-S. Kim, S. Lee, and N. Kwak. Dynamic graph generation network: Generating relational knowledge from diagrams. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4167–4175, 2018.
- [32] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.
- [33] I. Kokkinos. Highly accurate boundary detection and grouping. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2520–2527. IEEE, 2010.
- [34] J. Krishnamurthy, O. Tafjord, and A. Kembhavi. Semantic parsing to probabilistic programs for situated question answering. *arXiv preprint arXiv:1606.07046*, 2016.
- [35] J. Li, H. Su, J. Zhu, S. Wang, and B. Zhang. Textbook question answering under instructor guidance with memory networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3655–3663, 2018.
- [36] R. Li, S. Wang, F. Zhu, and J. Huang. Adaptive graph convolutional neural networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*,

2018.

- [37] Y. Li, W. Ouyang, and X. Wang. Vip-cnn: A visual phrase reasoning convolutional neural network for visual relationship detection. *arXiv preprint arXiv:1702.07191*, 2017.
- [38] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1261–1270, 2017.
- [39] X. Liang, L. Lee, and E. P. Xing. Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 848–857, 2017.
- [40] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- [41] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.
- [42] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.
- [43] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision*, pages 852–869. Springer, 2016.

- [44] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016.
- [45] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.
- [46] A. Micheli. Neural network for graphs: A contextual constructive approach. *IEEE Transactions on Neural Networks*, 20(3):498–511, 2009.
- [47] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.
- [48] T. Nguyen, M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, and L. Deng. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*, 2016.
- [49] M. Niepert, M. Ahmed, and K. Kutzkov. Learning convolutional neural networks for graphs. In *International conference on machine learning*, pages 2014–2023, 2016.
- [50] W. Norcliffe-Brown, S. Vafeias, and S. Parisot. Learning conditioned graph structures for interpretable visual question answering. In *Advances in Neural Information Processing Systems*, pages 8344–8353, 2018.
- [51] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 685–694, 2015.

- [52] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
- [53] J. Peyre, I. Laptev, C. Schmid, and J. Sivic. Detecting rare visual relations using analogies. *arXiv preprint arXiv:1812.05736*, 2018.
- [54] S. Qi, W. Wang, B. Jia, J. Shen, and S.-C. Zhu. Learning human-object interactions by graph parsing neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 401–417, 2018.
- [55] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.
- [56] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.
- [57] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NIPS*, 2015.
- [58] M. Richardson, C. J. Burges, and E. Renshaw. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, 2013.
- [59] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

- [60] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.
- [61] M. Shi, H. Caesar, and V. Ferrari. Weakly supervised object localization using things and stuff transfer. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3381–3390, 2017.
- [62] P. Tang, X. Wang, S. Bai, W. Shen, X. Bai, W. Liu, and A. L. Yuille. Pcl: Proposal cluster learning for weakly supervised object detection. *IEEE transactions on pattern analysis and machine intelligence*, 2018.
- [63] P. Tang, X. Wang, X. Bai, and W. Liu. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2843–2851, 2017.
- [64] P. Tang, X. Wang, A. Wang, Y. Yan, W. Liu, J. Huang, and A. Yuille. Weakly supervised region proposal network and object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 352–368, 2018.
- [65] D. Teney, L. Liu, and A. van den Hengel. Graph-structured representations for visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2017.
- [66] J. Uijlings, S. Popov, and V. Ferrari. Revisiting knowledge transfer for training object class detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1101–1110, 2018.
- [67] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *International journal of computer*



- vision*, 104(2):154–171, 2013.
- [68] P. Vicol, M. Tapaswi, L. Castrejon, and S. Fidler. Moviegraphs: Towards understanding human-centric situations from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8581–8590, 2018.
  - [69] W. Wang and J. Shen. Deep visual attention prediction. *IEEE Transaction on Image Processing*, 27(5):2368–2378, 2018.
  - [70] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. *arXiv preprint arXiv:1701.02426*, 2017.
  - [71] H. Xu and K. Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer, 2016.
  - [72] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 670–685, 2018.
  - [73] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 21–29, 2016.
  - [74] Z. Yang, D. Mahajan, D. Ghadiyaram, R. Nevatia, and V. Ramanathan. Activity driven weakly supervised object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2917–2926, 2019.
  - [75] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *2010 IEEE Computer Society*

- Conference on Computer Vision and Pattern Recognition*, pages 17–24. IEEE, 2010.
- [76] R. Yu, A. Li, V. I. Morariu, and L. S. Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1974–1982, 2017.
  - [77] Z. Yu, J. Yu, J. Fan, and D. Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 1821–1830, 2017.
  - [78] R. Zellers, Y. Bisk, A. Farhadi, and Y. Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6720–6731, 2019.
  - [79] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, 2018.
  - [80] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5532–5540, 2017.
  - [81] Y. Zhu, Y. Zhou, Q. Ye, Q. Qiu, and J. Jiao. Soft proposal networks for weakly supervised object localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1841–1850, 2017.
  - [82] B. Zhuang, L. Liu, C. Shen, and I. Reid. Towards context-aware interaction recognition for visual relationship detection. In *Proceedings of the IEEE*

*International Conference on Computer Vision*, pages 589–598, 2017.

## 초 록

이미지를 이해하는 것은 컴퓨터 비전 분야에서 가장 근본적인 목적 중 하나이다. 이러한 이해는 다양한 산업 분야의 문제를 해결 할 수 있는 혁신이 될 수 있다. 최근 딥러닝의 발전과 함께, 이미지에서 객관적인 요소를 인식하는 기술은 매우 발전되어 왔다. 그러나 시각 정보를 제대로 이해하기 위해서는 사람처럼 맥락 정보를 이해하는 것이 중요하다. 인간은 주로 직접적인 시각정보와 함께 맥락을 이해하여 의미 있는 지식 정보로 활용한다. 본 논문에서는 객체간의 의미적 관계정보를 구축하고 활용하는 방법론을 제시하여 보다 나은 이미지의 이해 방법을 연구하였다.

첫 번째로, 다이어그램에서 관계 지식을 표현하는 관계 그래프를 생성하는 알고리즘을 제안하였다. 다이어그램이 가진 정보를 축약하는 능력이 다른 형태의 지식 저장 방법에 비해 뛰어나지만, 그에 따라 해석하기에는 다양한 요소와 유연한 레이아웃 때문에 풀기 어려운 문제였다. 우리는 다이어그램에서 객체를 찾고 그것들의 관계를 찾는 통합 네트워크를 제안한다. 그리고 이러한 능동적인 그래프 생성을 위한 특수 모듈은 DGGN을 제안한다. 이 모듈의 성능을 나타내기 위해 모듈안의 활성화 게이트의 정보 역학을 비주얼라이즈하여 분석하였다. 또한 공개된 다이어그램 데이터셋에서 기존의 알고리즘을 뛰어넘는 성능을 증명하였다. 마지막으로 질의 응답 데이터셋을 이용한 실험으로 향후 다양한 응용 가능성도 증명하였다.

두 번째로, 우리는 현존하는 질의 응답 데이터셋 중 가장 복잡한 형태를 가진 교과서에서 질의응답 (TQA) 문제를 풀기위한 솔루션을 제안하였다. TQA 데이터셋은 질문 파트와 본문 파트 모두에 이미지와 텍스트 형태를 가진 데이터를 포함하고 있다. 이러한 복잡한 구조를 해결하기 위해 우리는 f-GCN이라는 다중 모달 그래프를 처리할 수 있는 모듈을 제안하였다. 이 모듈을 통해 보다 효율적으로 다중 모달을 그래프 형태로 처리하여 활용하기 쉬운 피쳐로 바꿔줄 수 있다. 그 다음으로 교과서의 경우 다양한 주제가 포함되어 있고 그에 따라 용어나 내용이 겹치지 않고 기술되어 있다. 그로인해 완전 새로운 내용의 문제를 풀어야하는 out-of-domain 이슈가 있다. 이를 해결하기 위해 정답을 보지 않고 본문만으로 자가 학습을 하는 알고리즘을 제안하였다. 이 두 알고리즘을 통해 기존 연구보다 훨씬 좋은 성능을 보이는 실험 결과를 제시하였고 각각의 모듈의 기능성에 대해 검증하였다.

마지막으로, 인간과 물건의 관계정보를 활용하여 객체 검출을 약지도 학습으로 배우는 프레임워크를 제안하였다. 객체 검출 문제를 풀기위해 노동력이 많이 필요한 데이터 라벨링 작업이 필요하다. 그 중 가장 노동력이 많이 필요한 위치 라벨링인데, 새로운 방법론은 인간과 물건의 관계를 이용하여 이부분을 해결하였다. 우리는 RRPN이란 모듈을 제안하여 인간의 포즈정보와 관계에 관한 동사를 이용하여 처음보는 물건의 위치를 추정할 수 있다. 이를 통해 새롭게 배우는 목표 라벨에 대해, 정답 라벨 없이 위치를 추정하여 학습할 수 있어 훨씬 적은 노력만 사용해도 된다. 또한 RRPN은 추가 방식의 구조로 다양한 태스크에 관한 네트워크에 추가 할 수 있다. HICO-DET 데이터셋을 사용하여 실험한 결과 현재의 지도학습을 대신할 가능성을 보여주었다. 또한 우리 모델이 처음 본 물건의 위치를 잘 추정하고 있음을 시각화를 통해 보여주었다.

**주요어:** 의미적 관계, 그래프 구조, 그래프 콘볼루션 네트워크, 객체 검출,  
약지도 학습

**학번:** 2014-24888

## 감사의 글

먼저 저를 제자로 받아주시고 석박사 통합과정을 무사히 끝낼수 있게 물심양면 지원해주신 곽노준 교수님께 감사드립니다. 교수님의 너그러운 지도 속에서 제 역량을 발휘할 수 있었던 것 같고 앞으로도 부끄럽지 않은 제자가 되도록 노력하겠습니다.

그리고 학위논문 심사에 참여해 주신 심사위원 교수님들께 감사드립니다. 제가 융합과학기술대학원에 입학할 수 있게 뽑아주시고 3학기 동안 지도해주셨던 서봉원 교수님께 감사드립니다. 또한 제가 개설하신 거의 모든 수업을 들었던 것 같은데, 냉철한 스마트함을 너무 닮고 싶은 이원종 교수님께도 감사드립니다. 그리고 심사기간 동안 진심어린 조언을 해주신 이교구 교수님과 멀리서 심사를 위해 찾아와주신 이민식 교수님께도 감사드립니다.

다음으로는 학위과정을 마칠 수 있게 많이 도와준 비빔두 식구들에게 감사드립니다. 상국, 명기, 규태, 성준, 경민를 비롯한 연구팀 그리고 경영지원팀과 데이터팀 등 모두 회사를 잘 이끌어 주셔서 제가 학위를 마칠 수 있지 않았나 싶습니다.

MIPAL랩 연구실 식구들도 감사드립니다. 논문 같이 쓴다고 고생했던 선훈, 규정, 정지수, 김지수 모두 감사하고 저의 첫 CVPR 논문에 큰 지도를 해주신 유영준 박사님에게도 감사를 드립니다. 랩을 옮겨온 후 초반에 많이 도와주었던 지용이 형님, 지은이, 혁진이, 헤민이, 지혜, 재영이, 상호, 지훈이형,

영규형, 한열이, 효진이 모두 감사드립니다. 그리고 제 뒤로 연구실에 와서 큰 역할을 해주는 시명이형, 승의, 장호, 재석이, 성욱이, 호준이 등 모두 좋은 연구 결과 만드시고 무사히 졸업하시기를 바라겠습니다.

우리 포스텍 1분반 친구들 그중 신박사, 성욱이, 승효 가끔 밥도 사주고 술도 먹어주고 해서 고마웠다. 그리고 은행 같이 그만두고 잘 챙겨주시는 준민이형님과 형수님께도 감사드립니다.

그리고 저희 가족들 모두 감사드립니다. 늦은 나이에 학위를 시작해서 걱정도 많이 하셨고 늘 저를 위해 기도해주시는 부모님께 무한한 감사들 드립니다. 회사 그만두는 사위에게 결혼을 허락해주신 장인어른과 장모님께도 감사드리고 앞으로도 더 좋은 모습 보여드리겠습니다. 저에게 많은 응원을 해준 동생과 태양이, 그리고 두 아이를 잘키우고 계신 손윗처남과 처남댁께도 그동안의 응원 감사드립니다.

마지막으로 지금 이 글을 쓰는게 믿겨지지 않는데, 이 길고 험난했던 모든 과정을 인내해주고 현실로 만들어 준 제 아내에게 이 학위를 바칩니다.